NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

USING MULTIPLE COLLABORATIVE AGENTS FOR ADAPTIVE QUALITY OF SERVICE MANAGEMENT OF C4ISR NETWORKS

by

Raymond A. Rivera

June 2001

Thesis Advisor:	Alex B. Bordetsky
Associate Advisor:	John S. Osmundson

REPORT DOCUMENTATION PAGE	*Form Approved OMB No. 0704-0188*

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.

1. AGENCY USE ONLY ***(Leave blank)***	**2. REPORT DATE** June 2001	**3. REPORT TYPE AND DATES COVERED** Master's Thesis

4. TITLE AND SUBTITLE: Using Multiple Collaborative Agents for Adaptive Quality of Service Management of C4ISR Networks	**5. FUNDING NUMBERS**
6. AUTHOR Raymond A. Rivera	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000	**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)	**10. SPONSORING / MONITORING AGENCY REPORT NUMBER**

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.	**12b. DISTRIBUTION CODE**

13. ABSTRACT ***(maximum 200 words)***

This research explores the potential of agent technology for adaptive Quality of Service (QoS) management of C4ISR networks. With the growing emphasis on information superiority, any time savings or additional utilization of resources enabled by effective network management becomes increasingly important. Intelligent agents are ideal for assessing information, adapting to dynamic conditions, and predicting future network conditions. In the kernel of the proposed multiple agent system (MAS) testbed are agent shared memory and majority rule architectures for agent conflict resolution. The case based reasoning (CBR) technique provides the foundation for building the agents' shared memory of QoS management solutions and allows the individual agents to share their associations of feedback controls in response to application and user QoS profiles. Based on the Telecommunications Management Network (TMN) functionality, we use this agent architecture to effectively translate the warfighter's service layer application requirements across the network. The fundamental frameworks of Service Level Management (SLM) and Policy Based Management (PBM) serve as cornerstones in effectively gathering and applying specific application requirements. Finally, we utilize these techniques to investigate an actual C4I application at the Pacific Region Network Operating Center (PRNOC) in Wahiawa, Hawaii as the real-world focal point of the thesis.

13. SUBJECT TERMS Command, Control, Communications, Computers, Intelligence, Surveillance, and Reconnaissance (C4ISR), Agent Technology, Quality of Service (QoS), Case Based Reasoning (CBR), Service Level Management (SLM), Policy Based Management (PBM), Telecommunications Management Network (TMN)	**15. NUMBER OF PAGES** 132
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT Unclassified	**18. SECURITY CLASSIFICATION OF THIS PAGE** Unclassified	**19. SECURITY CLASSIFICATION OF ABSTRACT** Unclassified	**20. LIMITATION OF ABSTRACT** UL

THIS PAGE INTENTIONALLY LEFT BLANK

USING MULTIPLE COLLABORATIVE AGENTS FOR ADAPTIVE QUALITY OF SERVICE MANAGEMENT OF C4ISR NETWORKS

Raymond A. Rivera
Lieutenant, United States Navy
B.S., United States Naval Academy, 1994

Submitted in partial fulfillment of the
requirements for the degrees of

**MASTER OF SCIENCE IN SYSTEMS TECHNOLOGY
(COMMAND, CONTROL, AND COMMUNICATIONS)**

MASTER OF SCIENCE IN INFORMATION TECHNOLOGY MANAGEMENT

from the

**NAVAL POSTGRADUATE SCHOOL
June 2001**

Author: ______________________
Raymond A. Rivera

Approved by: ______________________
Alex B. Bordetsky, Thesis Advisor

John Osmundson
____
John S. Osmundson, Associate Advisor

Dan C Boger
____
Dan C. Boger, Chairman
Computer and Information Systems/C3 Academic Groups

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

This research explores the potential of agent technology for adaptive Quality of Service (QoS) management of C4ISR networks. With the growing emphasis on information superiority, any time savings or additional utilization of resources enabled by effective network management becomes increasingly important. Intelligent agents are ideal for assessing information, adapting to dynamic conditions, and predicting future network conditions. In the kernel of the proposed multiple agent system (MAS) testbed are agent shared memory and majority rule architectures for agent conflict resolution. The case based reasoning (CBR) technique provides the foundation for building the agents' shared memory of QoS management solutions and allows the individual agents to share their associations of feedback controls in response to application and user QoS profiles. Based on the Telecommunications Management Network (TMN) functionality, we use this agent architecture to effectively translate the warfighter's service layer application requirements across the network. The fundamental frameworks of Service Level Management (SLM) and Policy Based Management (PBM) serve as cornerstones in effectively gathering and applying specific application requirements. Finally, we utilize these techniques to investigate an actual C4I application at the Pacific Region Network Operating Center (PRNOC) in Wahiawa, Hawaii as the real-world focal point of the thesis.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

I. INTRODUCTION 1

A. C4ISR IN THE 21ST CENTURY 1
B. SCOPE 4
C. EXPECTED BENEFITS OF THIS STUDY 5
D. OVERVIEW OF OTHER CHAPTERS 5

II. AGENT TECHNOLOGY 7

A. INTRODUCTION 7
B. AGENT TECHNOLOGY 8
1. What is an Intelligent Agent? 8
2. Agent Topology 9
3. Why Multiple? 10
4. Why Collaborative? 11
5. Why Adaptive? 12
C. CASE BASED REASONING 13
D. AGENT COMMUNICATION 14
E. AGENT ARCHITECTURE 15
1. Proposed Agent Architecture: The Committee Model/ANN 15
F. PROTEUS 18
1. Proteus Comparison Notes 20
G. GMD - GERMAN NATIONAL RESEARCH CENTER FOR INFORMATION TECHNOLOGY: SERVICE LEVEL MANAGEMENT WITH AGENTS 20
1. Zeus Agent Toolkit 22
2. GMD Comparison Notes 24
H. ATR COMMUNICATIONS RESEARCH LABORATORIES 24
1. ATR Comparison Notes 27
I. RETSINA 27
1. RETSINA Comparison Notes 29
J. HYBRID 30
1. Hybrid Comparison Notes 31
K. TELECOMMUNICATIONS MULTI-AGENT CONTROL SYSTEM (TELE-MACS) 32
1. Tele-MACS Comparison Notes 34
L. ACTS PROJECT: IMPACT 34
1. IMPACT Comparison Notes 37
M. SUMMARY 38

III. ADAPTIVE QOS MANAGEMENT 41

A. INTRODUCTION 41
B. SERVICE LEVEL MANAGEMENT 43
C. GATHERING REQUIREMENTS 44
1. Semantic Disparity Problem 45
2. Standard User Requirements 46

3. Application Requirements *47*
D. QUALITY OF SERVICE 48
1. Availability *48*
2. Throughput *49*
3. Packet loss *49*
4. Latency *49*
5. Jitter *49*
E. POLICY BASED MANAGEMENT 52
1. General Technology Requirements *53*
a. Service Differentiation 53
b. Network Provisioning and Bandwidth Management 53
c. Integration with Network Management Systems and Legacy Devices 54
d. Scalability 54
e. Industry Standardization 54
f. Security 54
F. TELECOMMUNICATIONS MANAGEMENT NETWORK (TMN) 54
G. DEVELOPING THE PROPOSED AGENT FRAMEWORK 56
1. Layers of Feedback Control: Individual Agent Adaptation *57*
a. Call Preparation Control Requirements 57
b. Connection Control Requirements 58
2. Call Preparation Adaptation: Service Layer Feedback Controls *58*
3. Connection Control Adaptation: NE Layer Adaptation *61*
H. SUMMARY 62

IV. PACIFIC REGION NETWORK OPERATIONS CENTER (PRNOC) 63

A. PACKETSHAPER 63
1. Packetshaper Key Features [Performance Specification]: *65*
2. Packetshaper Sizing Considerations [PRNOC CONOPS] *65*
3. Accessing the Packetshaper *66*
B. NOC ENVIRONMENT – BACKGROUND 66
1. Automated Digital Network System (ADNS) *67*
C. BASIC BANDWIDTH MANAGEMENT POLICIES 68
1. For ADNS Circuits, Keep (UNCLAS) MSS Size at 1200 or Below *68*
2. Do Not Exceed Circuit Capacity *69*
3. Provide Enough Bandwidth for Mail *69*
4. Provide a Nominal Bandwidth for DNS *69*
5. Allow Remainder of Available Bandwidth for Webbing *69*
6. Restrict Lotus Domino Bandwidth to Subs (GENSER) *69*
7. Dampening: FTP, RealAudio, RealVideo *69*
D. MONITORING/SHAPING 70
1. Monitoring *70*
2. Shaping *71*
F. USING PARTITIONS TO IMPLEMENT THE POLICIES 71
1. Do Not Exceed Circuit Capacity *71*

2. *Smaller Bandwidths* 72
3. *Small Decks* 73
4. *Inbound/Outbound Partition Sizes* 74
5. *Providing Enough Bandwidth for Mail (SMTP)* 74
6. *Burstable* 75
7. *Provide Nominal Bandwidth for DNS* 75
8. *Allow Remainder of Bandwidth for Webbing* 75
9. *Restrict Lotus Domino Bandwidth to Subs (GENSER)* 75
10. *Dampening: FTP, RealAudio, RealVideo* 75

F. OTHER CONSIDERATIONS 76
1. *Ships with Multiple Servers* 76
2. *Ships with Multiple Links* 76
3. *Changing Bandwidth* 76

V. AN AGENT SOLUTION FOR PRNOC 77

A. PRNOC QOS DILEMMA 77
B. DATA 80
1. *Data Limitations* 80
2. *Data Trends* 81

C. AGENT SOLUTION 82
1. *Methodology* 82
2. *Learning* 83
3. *Deriving Agent Committees* 84

D. AGENT IMPLEMENTATION 85

VI. CONCLUSIONS AND RECOMMENDATIONS 87

A. SUMMARY 87
B. CONCLUSION 89
C. RECOMMENDATIONS FOR FURTHER STUDY 89
1. *Continue Development of an Agent Solution at PRNOC* 89
2. *Continue Bandwidth Usage Study at PRNOC* 90
3. *Modeling and Simulation* 90
4. *Develop an Agent Testbed* 91
5. *Explore Wireless Feasibility Issues* 91

APPENDIX. BANDWIDTH UTILIZATION DATA 93

LIST OF REFERENCES 111

INITIAL DISTRIBUTION LIST 115

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGEMENTS

First, I would like to acknowledge the financial support of the Joint Experimentation Directorate, U. S. Joint Forces Command (J-9, JFCOM), which made possible the data collection and research at the Pacific Region Network Operations Center in Wahiawa, Hawaii. Thanks to Lt. Paul Carff and Mr. Wren Lee of PRNOC for their assistance and helpfulness.

Next, I would like to thank Professor Bordetsky for his nearly yearlong guidance and teaching. It was wonderful being able to tap into his knowledge and expertise in agent technology and networks. Thanks also to Professor Osmundson, whose vast industry and teaching experience made him an invaluable resource in finishing off the thesis. I am thankful to both for their time and patience.

Finally, thanks to my wife, Tsugumi for putting up with all my hours of studying in only our first year of marriage. Her patience, understanding, and company were a great comfort and source of strength. We made a great team and made the best of our time together in Monterey.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. C4ISR IN THE 21ST CENTURY

C4ISR networks of the future are increasingly reliant on fast, efficient information exchange over wide distances. In the 21st century, information superiority is the key to battlespace dominance. C4ISR networks are the enablers to this goal and critical in the Navy's movement towards Network Centric Warfare. At a minimum, C4ISR networks must be capable of providing voice, video, and data capabilities to the warfighter. At the same time, the information exchange must be accurate, timely, and secure in order to be useful. These factors make the effective management of C4ISR networks paramount. As the growth of information technology increases, so does the need for coordination and maintenance.

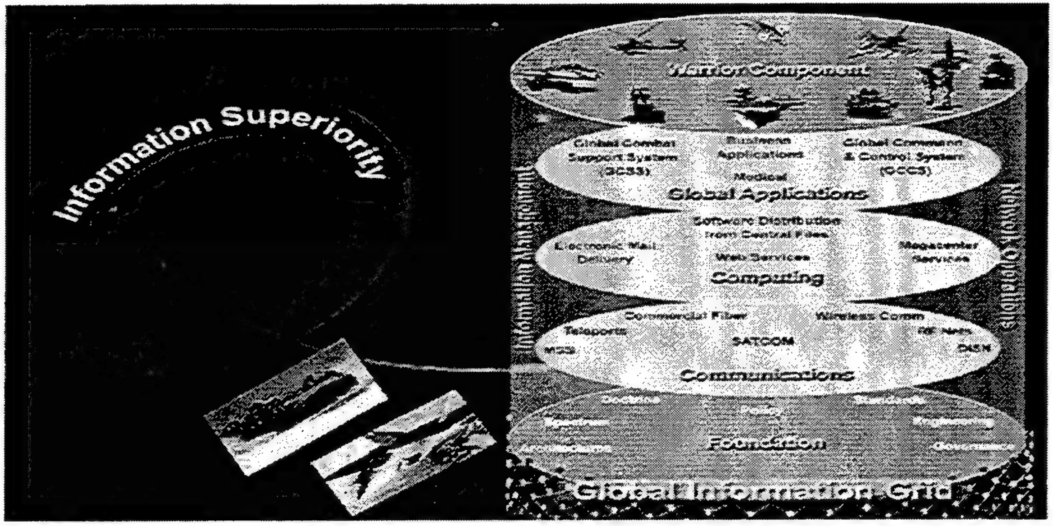


Figure 1.1. Joint Vision 2020. From [JV2020].

The evolution of C4ISR networks and their management systems over the years has resulted in a variety of network management issues. Even though all joint C4ISR networks are supposed to follow the same basic guidelines and interoperability standards under the Joint Technical Architecture (JTA) and Defense Information Infrastructure Common Operating Environment (DII-COE), this is not always the case due to the difficulty in tying in the broad range of legacy C4ISR applications. The problems of: diverse services, networks, and technologies; multiple vendor equipment; loosely organized management applications; multiple management protocols; and multiple data representations all have a direct impact on network management and quality of service (QoS) [Bieszczad et al]. In the final analysis, C4ISR networks must be capable of adapting end-to-end resources and QoS across heterogeneous, and oftentimes, mobile networks.

In general, management of these networks occurs at Network Operations Centers (NOCs). NOCs utilize standard network monitoring approaches like Simple Network Management Protocol (SNMP) and Common Management Information Protocol (CMIP) to monitor, test, and evaluate network parameters including traffic patterns, bandwidth utilization, network response times, and e-mail response times. Unfortunately, with increasing requirements for fast and efficient information exchange, these techniques need improvement and adaptive management capability.

Adaptive management capability for C4ISR networks could be achieved through the usage of multiple collaborative, intelligent agents to overcome the deficiencies in C4ISR network management. Although agent technology has only recently gained prominence in the last ten years, it has already demonstrated exciting potential in a variety of applications that lend themselves to this research. Basic agent characteristics of *autonomy, adaptability, scalability,* and *cooperability* allow the sharing of information over the entire span of the network. Intelligent agents assess information, adapt to existing conditions, predict future network conditions, and advise on anticipated future conditions. With multiple, collaborative agents, knowledge and expertise can be shared, eliminating the need to store all necessary knowledge locally. In the context of a dynamic environment with unique application profiles, this framework is ideally suited for translating the warfighter's service level requirements. The end result is a more efficient, responsive, and potent C4ISR network.

In the kernel of the proposed multiple agent adaptive management testbed are agent shared memory and majority rule architectures for agent conflict resolution. The *case based reasoning* (CBR) technique will be used as the foundation for building the agents' shared memory of QoS management solutions. It allows the individual agents to share their associations of feedback controls in response to application and user QoS profiles.

The *committee* type multi-participant group decision support technique will be adopted for resolving the conflicts among multiple agents in allocating the networking resources in response to the conflicting QoS requirements. The conflict resolution architecture is composed of an artificial neural network (ANN) with two hidden layers. Each node in the second hidden layer represents the committee solution for QoS resources allocation that the multiple agent system (MAS) learned while managing the C4ISR task and adapting to the conflicting QoS requirements.

In accordance with the Telecommunications Management Network (TMN) functionality, the agent architecture effectively translates the warfighter's service layer application requirements across the network. The fundamental frameworks of Service Level Management (SLM) and Policy Based Management (PBM) are the cornerstones in effectively gathering and applying specific application requirements. From these requirements, the multiple agent testbed becomes the enabling framework for the intelligent adaptive capability of collaborative work.

Using these building blocks for our research, we investigate an actual C4I application at the Pacific Region Network Operations Center (PRNOC) in Wahiawa, Hawaii and use it as the real-world focal point for this thesis. In this particular instance, we investigate the adaptive allocation of bandwidth under dynamic conditions via multiple collaborative agents.

In sum, this research will develop the potential of agent technology for the efficient management of C4ISR networks. With the growing emphasis on information superiority, any time savings or additional utilization of scarce resources enabled by effective network management could be the difference between victory and defeat for the warfighter. C4ISR communications must be Robust, Reliable, Redundant, and Ready (4R's) [Seventh Fleet]. Agent technology can be an answer to these requirements.

B. SCOPE

The scope of this thesis includes: (1) an in-depth review of agent technology including characteristics, functions, collaboration techniques and agent architectures; (2) a review of fundamental network management concepts including SLM, TMN, QoS, and

PBM; (3) a survey of network data at the Pacific Region Network Operating Center (PRNOC) to develop specific application trends and requirements; (4) a feasibility study of implementing agent technology for a representative C4ISR application at PRNOC; and (5) a concluding feasibility study of how agent technology may serve as an improvement in QoS management. The thesis will conclude with a recommendation for transitioning current C4ISR architectures to include agent technology for QoS adaptation in network management

C. EXPECTED BENEFITS OF THIS STUDY

This project is a first-time study into the effectiveness of using multiple collaborative agents for QoS adaptation in C4ISR networks and highlights an actual C4I application to investigate the feasibility of implementing agent technology. The project provides background for the developing agent-based network management testbed at the Naval Postgraduate School and will serve as an example for other DoD organizations.

D. OVERVIEW OF OTHER CHAPTERS

This chapter is an introduction to the research covered in this thesis. In Chapter II, we take an initial look at agent technology. The usage of the term "agent" is defined for the context of this thesis. We analyze the suitability of the case based reasoning learning technique for agent adaptability in a dynamic environment and evaluate various agent architectures for suitability to the research task, with particular emphasis on the proposed ANN framework.

In Chapter III, we progress into the usage of agents for adaptive QoS management in C4ISR networks. The underlying concepts of Service Level Management (SLM), Telecommunications Management Network (TMN), Quality of Service (QoS), Policy Based Management (PBM), and requirements gathering are discussed.

In Chapter IV, we utilize these concepts to acquire information at the Pacific Region Network Operations Center (PRNOC) in Wahiawa, Hawaii. From the empirical data, we develop specific application requirements to be intelligently managed by agents.

In Chapter V, we investigate the proposed agent architecture and its suitability in the PRNOC C4ISR network architecture. Real application requirements and operating principles gathered at PRNOC are used as the basis for the model. The results of the chapter include a potential agent framework for specific usage at PRNOC.

Chapter VI contains the final conclusions of this research, a feasibility recommendation of agent technology for adaptive QoS management, and recommendations for further study.

II. AGENT TECHNOLOGY

In this chapter, we examine agent technology, and, in particular, "*multiple*", "*collaborative*", and "*adaptive*" agents. Each of these descriptors has a distinct meaning with respect to agent technology and plays an important role in the chosen task of adaptive QoS management. Subsequently, we review several candidate multi-agent system (MAS) architectures for suitability in the research and study in greater detail a proposed architecture based on case based reasoning (CBR), the committee decision approach, and an artificial neural (ANN) network architecture.

A. INTRODUCTION

Agent based technology is an interdisciplinary area of research that started receiving special attention from the research community in the early 1990's. This technology demonstrates exciting potential for the artificial intelligence (AI) and computer science communities because of its ability to reach a broad range of applications across many industries. To reach this potential, there are also many challenging problems including *security*, *resource consumption*, *complexity*, and the degree of *trust* in agents to do exactly what is desired. While these challenges are real, they are not enough to dampen the spread of the agent paradigm. Researchers are continually developing innovative new approaches to agent technology.

From DoD's perspective, agent technology is expected to help reduce time spent manipulating stovepipe command and control (C2) systems, make it easier to assemble future systems, improve interoperability, reduce system complexity, and help solve data blizzard and information starvation problems [Manola & Thompson]. Agent applications range from robotics to information retrieval to e-commerce to network management and telecommunications. Based on its wide range of applicability, it is very plausible that agent technology can be effectively utilized for adaptive QoS management.

B. AGENT TECHNOLOGY

1. What is an Intelligent Agent?

In general, intelligent software agents are a relatively new class of software that act on behalf of the user to find and filter information, negotiate for services, easily automate complex tasks, or collaborate with other software agents to solve complex problems. The main idea behind software agents is *delegation*, whereby the user delegates a task to the agent. In turn, agents act *autonomously* to perform the task on behalf of the user. In order to facilitate task accomplishment, *communication* is an important interface between user-to-agent and agent-to-agent. Finally, the agents must be able to *monitor* the state of their environment and make the decisions necessary to complete their tasks. [AgentBuilder]

When working with agent technology, the first order of business is effectively localizing the meaning of the term "agent," for there are literally hundreds of definitions and contexts. The term agent is highly overused and can mean different things to different applications. For instance, in network management, there are SNMP and CMIP agents, but these are really nothing more than servers providing data to their clients. On the other hand, there are expert systems with huge knowledge bases, which are also considered agents because of their intelligent behavior. This thesis focuses on the latter type of agent that intelligently makes decisions. Ultimately, these agents interface with the SNMP/CMIP agent functionality only as the abstraction of higher-level requirements to lower level requirements on the network layer.

In general, the following basic definition of agent applies to this thesis: "A computational entity that acts on behalf of others; is *autonomous*, *adaptive*, and *intelligent*; and exhibits the ability to learn and cooperate (*collaborate*) " [Bieszczad et al, p. 116]. More advanced agents may also have other attributes, such as *mobility* (allowing migration from host to host) and *personality* (manifesting some human qualities such as cooperation, caution, and greed). These additional characteristics can be explored as possible enhancements to the research.

2. Agent Topology

As for the classification of agents, the range of methods to develop a standard topology is highly varied. One prevalent method of classifying agents is in terms of dimensions. Certainly agents cannot only be described in just two or three ways because of the variability of the term and the need to accurately distinguish one agent from the next. In accordance with this fact, agents can first be classified by their *mobility*, i.e., by their ability to move around some network. Thus, they may be classified as *static* or *mobile*. Second, agents can be classified by the presence of a symbolic reasoning model, as either *deliberative* or *reactive*. Deliberative agents engage in planning and negotiation with other agents to achieve goals, while reactive agents respond to the present state of the environment in which they are a part. Third, agents can be classified by the exhibition of ideal and primary attributes such as *autonomy, learning*, and *cooperation* to derive the following four types of agents: *collaborative, collaborative learning, interface,* and *truly smart* agents (Figure 2.1). Fourth, agents may be classified according to their roles such as *information* or *Internet* agents. Fifth, agents can be classified as *hybrid* if they combine two or more agent philosophies in a single agent. Lastly, agents may exhibit any of a wide range of secondary attributes. In sum, just as the means of defining agents is diverse, so are the methods of classifying them. [Nwana]

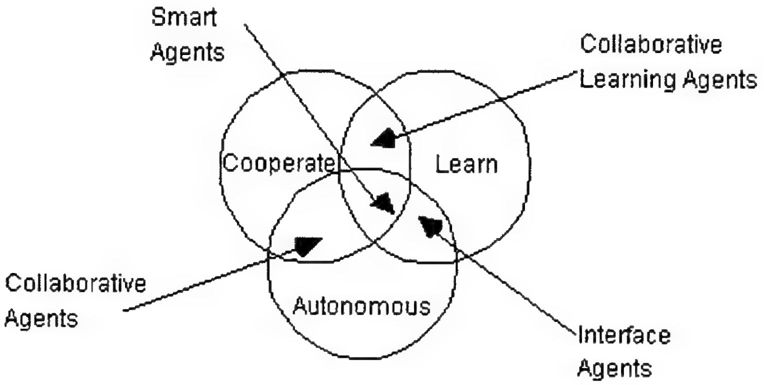


Figure 2.1. Topology Based on Primary Attribute Dimension. From [Nwana].

3. Why Multiple?

There are many reasons why a multi-agent approach is more advantageous than a single agent approach. First of all, the management of C4ISR networks is too large a problem for a single centralized agent. There are resource limitations and robustness concerns in only using a single agent. Decentralization takes away the possibility of a single point of failure. Moreover, dividing functionality among agents provides modularity, flexibility, modifiability, and extensibility [Green Paper]. Second, multiple agents allow for the interconnection of multiple existing legacy systems, which can be especially helpful in DoD. By building an agent wrapper around such systems, they can be incorporated into an agent society. Third, multiple agents improve scalability due to the organizational structure of the agents, which allows them to dynamically change to reflect the dynamic environment. Fourth, multiple agents provide solutions for inherently distributed problems by drawing from distributed information sources and distributing the expertise. For these reasons, multi-agent systems are more prevalent than single agent systems. [Hayzelden & Bigham]

4. Why Collaborative?

The collaborative behavior criterion for intelligent agents coincides with social ability. By *collaborative*, the usage of a multiple agent system is implied. Collaborative agents work in concert with other agents to achieve a common goal. The rationale for having collaborative agent systems is a specification of the goal of distributed artificial intelligence (DAI). It may be stated as: "creating a system that interconnects separately developed collaborative agents, thus enabling the ensemble to function beyond the capabilities of any of its members" [Nwana & Ndumu, Sec. 5.1.1]. The criterion of "collaborative" goes hand in hand with "multiple" in that it dictates teamwork among the agents. Agents cannot be collaborative without other agents to collaborate with. In other words, the union of the two characteristics is integral to the accomplishment of the factors listed above.

When considering the usage of collaborative agents, there are many factors to consider. The first problem is engineering the construction of collaborative agent systems by moving away from "point solutions to point problems" [Nwana & Ndumu]. This entails using methodologies that allow for quicker and more structured implementation of multi-agent systems. The second problem is inter-agent coordination, in which the concern is to effectively solve problems with certain constraints in resource boundedness and time. Third, *stability*, *scalability*, and *performance* must obviously be accounted for. Fourth, the learning mechanisms must be examined, whether they be machine learning, case based reasoning, etc. Fifth, there must be a means to verify and validate the collaborative agent systems meet their functional specifications. [Nwana & Ndumu]

5. Why Adaptive?

An agent is considered *adaptive* if it is capable of responding to other agents and/or its environment to some degree. At a minimum, the agent must be able to react to a certain stimulus. For this research, adaptive also means the ability to reason, learn, and evolve. These agents are deliberative and can change their behavior based on experience and a dynamic environment. Learning techniques include artificial neural networks, Bayesian rules, credit assignments, classifier rules, and case based reasoning. Adaptive agents can be *passive*, whereby they respond to environmental changes without attempting to change the environment; or *active*, whereby they exert some influence on the environment to improve their ability to adapt.

Unfortunately, by providing agents with the capability to adapt, there is also a possibility of inducing undesirable side effects – particularly in situations where global system behavior may be significantly affected by a minor local change [Gordon]. An adaptive agent must be able to adapt to unforeseen conditions, have a reasonable amount of behavioral assurance, and be able to respond in a timely manner. When developing adaptive agents, one must consider the tradeoff between verification of proper agent coordination and speed. If the agents cannot act in a fast enough manner, this obviously defeats the purpose of having them. Despite this dilemma, the characteristic of adaptability remains integrally important in allowing the agents the ability to respond and thrive in dynamic environments.

C. CASE BASED REASONING

As stated in the introduction, we focus on the case based reasoning (CBR) approach to problem solving and apply it to the agents' learning process. In the kernel of the proposed intelligent support architecture is the layered model of case memory. Case memory is useful in that it supports the discovery of pertinent collaborators, the retrieval of information pertinent to collaboration, and the creation of conventions among individuals by utilizing the CBR technique for indexing, capturing, and retrieving collaborative objects.

As a source of comparison, the logic behind CBR usage is similar to the usage of case law in the legal domain. In this domain, case studies are used as a point of reference. Lawyers and judges examine pre-existing case law to determine applicability to current cases at hand. Of course, not every new case is exactly like an old one, but the advantages of being able to apply prior work and experience to a new situation are clear. Not having to "reinvent the wheel" every time alleviates the amount of work to be done, while simultaneously giving higher credence to the ultimate outcome of the case.

The general architecture for CBR illustrates the evolutionary nature of the case library. In the *retrieve* stage, case law is injected into the process as an initial step in determining similarity with the current input. Next, in the *adaptation* stage, the system attempts to reconcile case memory with the new situation. *Execution* follows and the case library is updated with the new method in the *organization* phase. In this manner, the knowledge base is continually updated. [Lewis 1995]

D. AGENT COMMUNICATION

Communication is the backbone to any agent system because it allows agents to share information and thereby determine the overall behavior and organization of the system. Agent communication is accomplished with three components: *ontology*, *content language*, and *agent communication language* [Biescszad et al]. *Ontology* is a collection of terms and rules that define, govern and localize a certain domain. The *content language* is used for information encoding through statements about the domain, which combine terms from the corresponding ontology into meaningful sentences. The *agent communication language* (ACL) provides formalism for exchanging messages.

Currently, agent communication is one of the most important areas for standardization. The Object Management Group (OMG) is one agency attempting to ensure the variety of communication languages is kept at a minimum. Messages must have a well-defined semantics that is computational and visible. Therefore, ACLs are required for interoperability. ACLs must have formal semantics so that different implementations preserve the essential features. Possible implementations include:

- Knowledge Query Manipulation Language (KQML)
- Foundation for Intelligent Physical Agents (FIPA) ACL
- Knowledge Interchange Format (KIF)
- XML-based

There are two standards regarding agent-based systems: FIPA ACL and OMG's Mobile Agent System Interoperability Facilities (MASIF). The interactive nature of multi-agent systems drives the need to support interoperability between agents from various sources. Moreover, the development of such a standard is necessary for the successful utilization of agent technology in an open environment.

E. AGENT ARCHITECTURE

In this and subsequent sections, we highlight various prospective agent architectures that might be suitable to this kind of research. In particular, we focus on the Multi-Agent System approach and do not consider other approaches such as mobile agent, ant-based, or economic.

Due to the limited time and scope of this thesis, we provide a more direct focus on only the *committee model/artificial neural network* in order to provide a better, more in-depth look at our proposed candidate architecture. As concepts are discussed in subsequent chapters, the model is further developed until the final model incorporates the ideas of case based reasoning, the committee decision-making approach, an artificial neural network design, and adaptive QoS management capability.

1. Proposed Agent Architecture: The Committee Model/ANN

In practice, the collaborative multiple agent architecture will be used in conjunction with network operations management teams decision support relationships. Therefore, we consider the perspective collaborative multiple agent structures using the multi-participant information processing and networking paradigm. In accordance with this paradigm, decision-making relationships can take place locally or span across vertical and horizontal organizational boundaries. In turn, standard network computing topologies can be applied to derive the three basic models of *group, team*, and *committee*. [Bordetsky].

In the group model (Figure 2.2), the structure of information flows is a mesh network that links multiple decision-makers in a way that allows complete interaction among them.

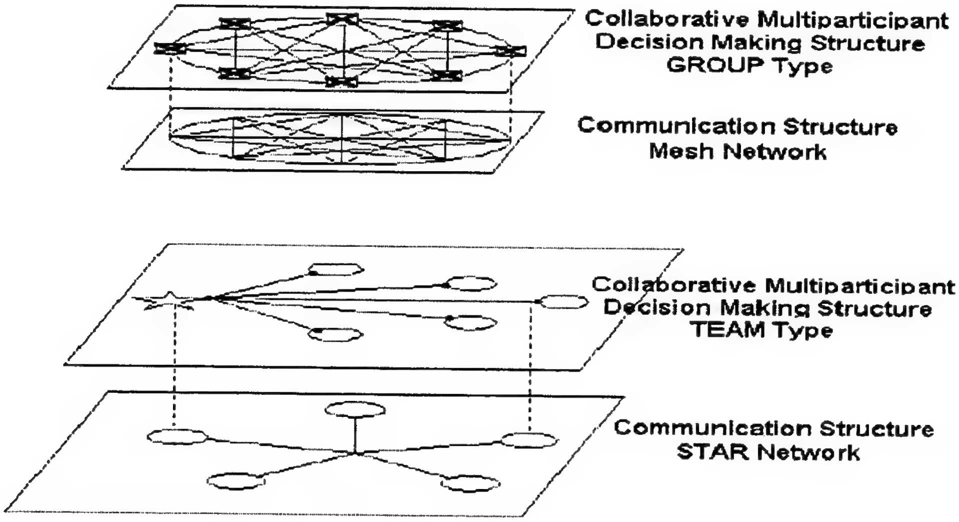

Collaborative Multiparticipant
Decision Making Structure
GROUP Type
Communication Structure
Mesh Network
Collaborative Multiparticipant
Decision Making Structure
TEAM Type
Communication Structure
STAR Network

Figure 2.2. Group and Team Type Multi-participant Structures. From [Bordetsky].

The team model represents a more centralized pattern of a single decision-maker with no participant interaction. Several local area and wide area communication topologies could satisfy the team structure support requirements. The primary topology is generally star and fits local and interdepartmental relationships. Also, bus and ring provide chain and circle type relationships to the team members.

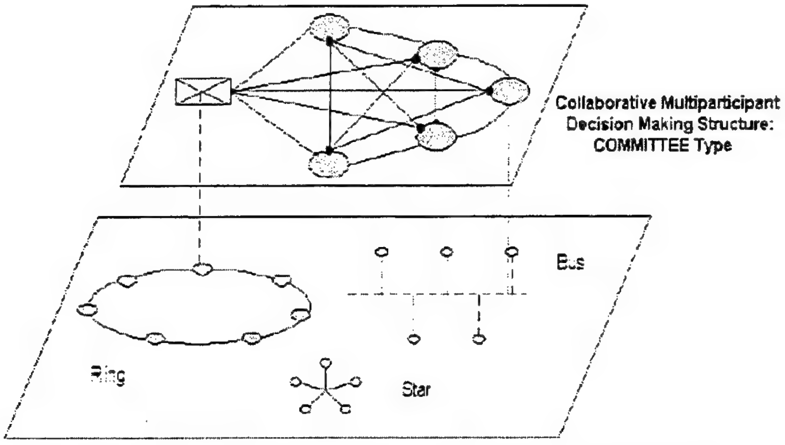


Figure 2.3. Committee Structure. From [Bordetsky].

The third basic model is committee (Figure 2.3) and is composed of multiple levels. This model combines a single decision-maker on the first level with the complete participant interaction on the next. In turn, this allows collective behavior that is based on the different types of majority rules or consensus protocols. On the second level, a combination of star and ring topologies could be used to support local and interdepartmental committee structures.

To summarize, *group* multi-participant structures may not be the most appropriate prototype for the multiple agent adaptation since they rely on the mesh topology and do not separate the facilitator (coordinator) from the other members. Unlike it, the *team* topology naturally allocates a role for the decision-maker (facilitator), but lacks cooperative relationships among the members, which is critical in the joint knowledge discovery process. For these reasons, the *committee* model represents the best compromise between the group and team multi-participant structures. In other words, it allows a facilitator (coordinator) role, while at the same time compensating for the lack of participants' interaction that is typical for the team structure.

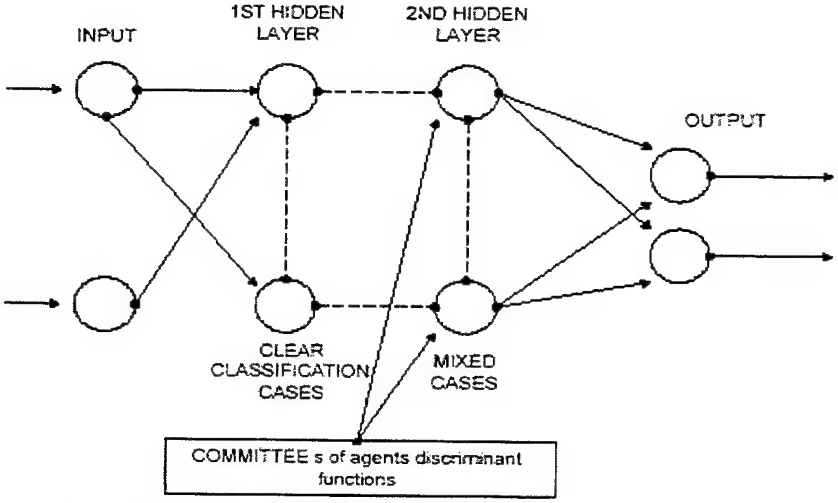

INPUT
1ST HIDDEN LAYER
2ND HIDDEN LAYER
OUTPUT
CLEAR CLASSIFICATION CASES
MIXED CASES
COMMITTEE s of agents discriminant functions

Figure 2.4. Representing the Agent Committees in an Artificial Neural Network. From [Bordetsky].

With respect to adaptive QoS, the agent committees will be implemented in a four-step artificial neural network as shown in Figure 2.4. The layers consist of an input layer, first hidden layer, second hidden layer, and output. The first hidden layer of agents resolves relatively easy cases to allow for network bandwidth adaptation without any contradiction. The second hidden layer resolves more challenging cases. In this layer, the selection criteria for the committee of constraints may vary. When considering factors that are all considered equal, the selection criterion is a simple majority rule. The learning process will compare the new problem with the set of developed (learned) empirical constraints that represent the network layer bandwidth adaptation experience (case memory).

F. PROTEUS

Proteus is a multi-agent system prototype being implemented as part of British Telecommunications' (BT) Intelligent Network Management Research Program to minimize the number of rejected calls and to maximize network resource utilization within an ATM network. Proteus uses collaborative intelligent agents to acquire large amounts of data in real-time from distributed ATM elements, assess the information, predict future network conditions, and advise on anticipated future conditions. [Odubiyi et al]

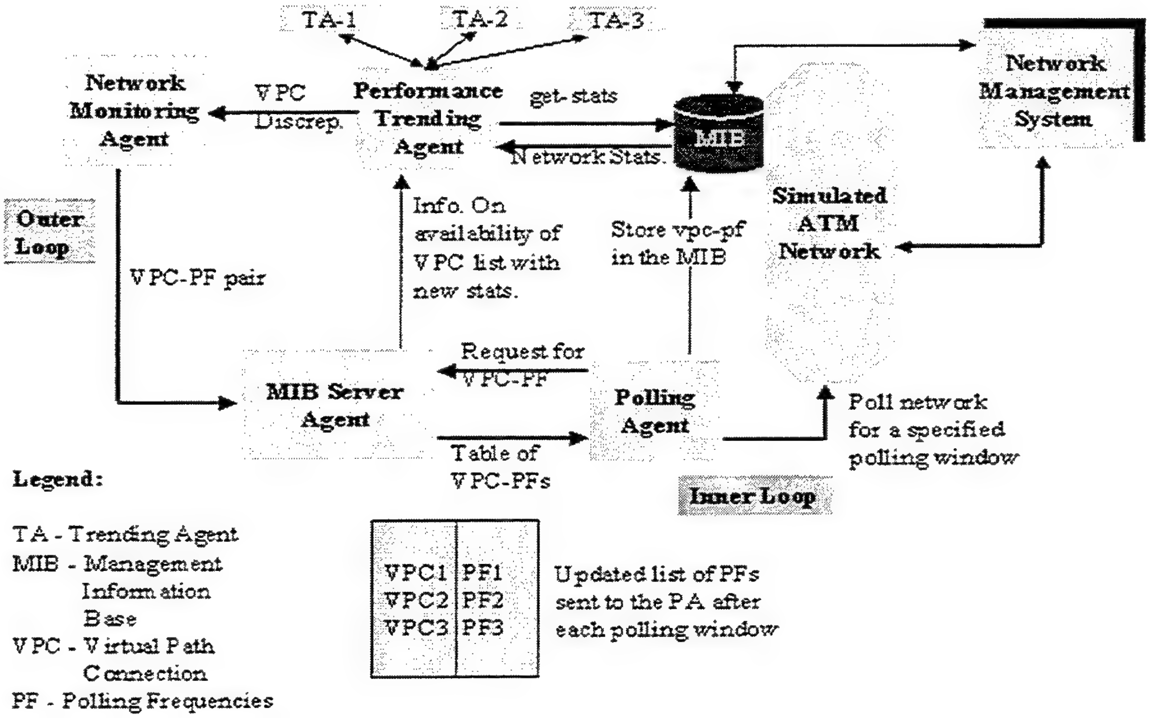


Figure 2.5. Proteus: MAS Interaction Diagram and Network Polling Process. From [Odubiyi].

Figure 2.5 shows the Proteus multi-agent collaboration process. The *Polling Agent (PA)* polls the network virtual path connection (VPC) for a specified polling window. Polled statistics are stored in the management information base (MIB). The *Performance Trending Agent Manager (PTAM)* retrieves the network usage statistics and delivers them to three *trending agents (TAs)* with a request to compute the minimum discrepancy between the actual VPC bandwidth usage statistics and expected usage.

Each TA employs a separate trending algorithm to compute network usage discrepancy and returns them to the PTAM. The PTAM selects the lowest discrepancy value and relays it to the *Network Monitoring Agent (NMA).* The NMA uses the discrepancy value to calculate a new network usage polling frequency (PF) for specific VPCs. The VPC-PF pairs are forwarded to the *MIB Server Agent (MSA)* that maintains a table of VPC-PF pairs. At the end of a polling cycle the PA retrieves a new set of VPC-PF information from the MSA. The MSA also provides VPC usage statistics to the PTAM for use by the TAs. [Odubiyi et al]

1. Proteus Comparison Notes

Although Proteus was intended for adaptive network management, it was not necessarily designed for translating service layer requirements. Instead, Proteus is currently used for adaptive variable polling and operates on a lower layer. But based on the effectiveness of the agent structure and the program's initial success, it is conceivable that this framework can be leveraged for future developments that coincide with the purposes of this thesis. The Proteus architecture shows a good working interface with the Management Information Base (MIB). As for the differences, the Trending Agents do not collaborate in arriving at decisions. Moreover, there is no hierarchical voting structure.

G. GMD - GERMAN NATIONAL RESEARCH CENTER FOR INFORMATION TECHNOLOGY: SERVICE LEVEL MANAGEMENT WITH AGENTS

In GMD's setup, a user explains his request for service level management to the Interface Agent (Figure 2.6). The *Interface Agent* helps decide what kind of service should be evaluated, locate services, and define boundary conditions. For example, if a user were bound by a Service Level Agreement (SLA), the user would set measurement values accordingly, explain frequencies of notifications by the Interface Agent, and determine other visualization options. In turn, the Interface Agent would report and visualize results. Because the Interface Agent is only responsible for adopting the task to the user's orders and presenting results, it gives birth to an *Agent Manager* and delegates the boundary conditions, configuration and the given task to it. This Agent Manager is the highest agent in the hierarchy apart from the Interface Agent. While the Interface Agent waits for results, the Agent Manager builds up an agent society by applying to the registry, which has a stock of agents with different characteristics. The registry knows possible platforms for agents with actual resources. The Agent Manager selects appropriate task agents and builds agent teams as necessary. These teams are given a certain competence that has influences on their decision power. In performing its task, each agent can decide to become an Agent Manager in the borders of its competence. Each agent except for the Interface Agent plays the role of a Task Agent being configured and supervised by its Agent Manager and it can act as an Agent Manager itself by building subordinate agent teams. The agents do their job on their platforms, notify their Agent Manager if necessary and communicate with their team agents. If an Agent Manager kills one of his agents because it is no longer used, it will inform the platform of the arising resource, which will keep the registry up to date. Platforms also can resign or take part in agent projects in communicating with the registry. [Bissel et al]

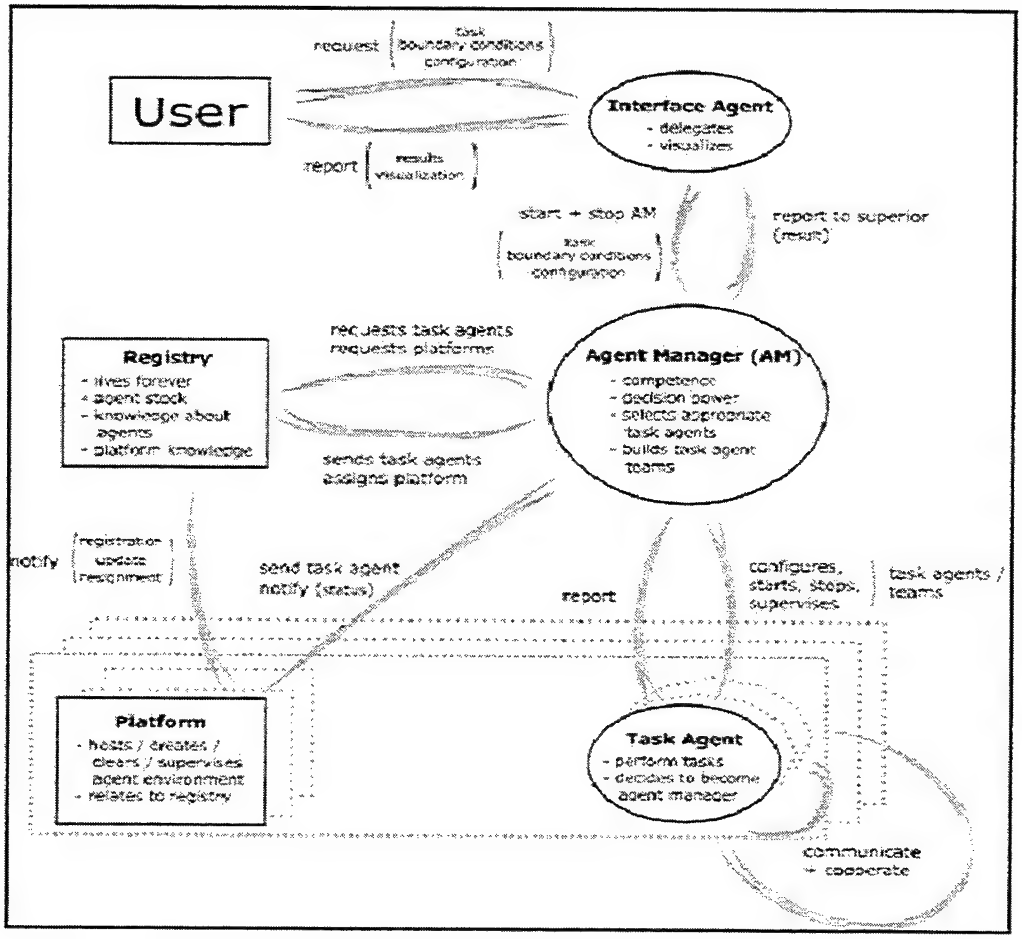


Figure 2.6. GMD Agent Workflow. From [Bissel et al].

1. Zeus Agent Toolkit

While not an agent architecture in itself, ZEUS is noteworthy as the proposed agent toolkit for the GMD project. The ZEUS Agent Toolkit is ideal for utilizing heterogeneous autonomous agents for collaboration in solving large-scale problems and is the culmination of a careful synthesis of established agent technologies to provide an integrated environment for the rapid development of multi-agent systems (Toolkit). Figure 2.7 is a context diagram illustrating some of the issues involved in knowledge level multi-agent collaboration. The *Central Agent* needs to perform a complex task that requires it to collaborate with other agents. To do so, it uses the *Facilitator* to discover the agents with the required abilities, and the *Agent Name Server* to determine the addresses of these agents. The inter-agent communication language is used to communicate with the Agent Name Server, Facilitator, and other agents and requires a shared representation and understanding of common domain concepts. [Collis & Ndumu]

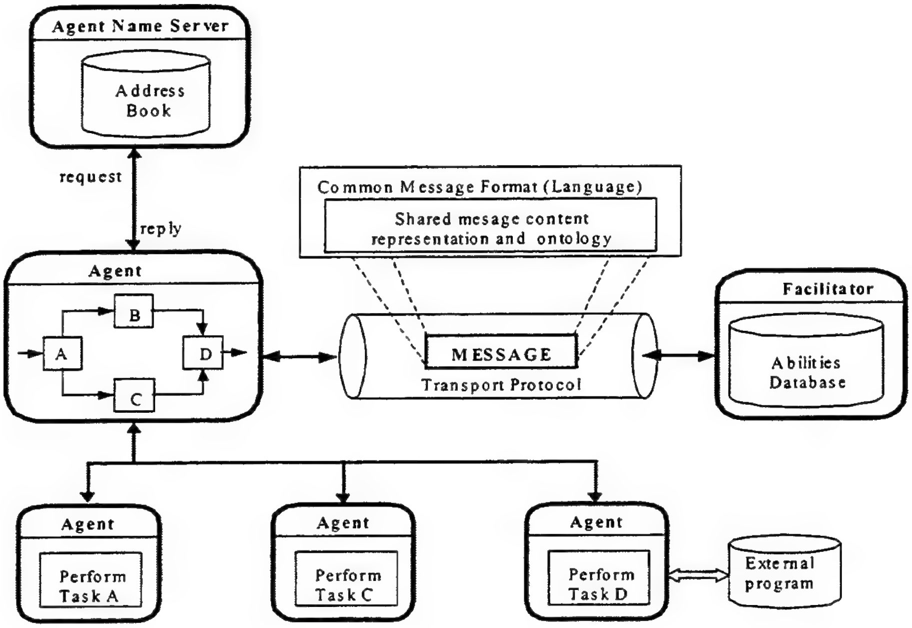


Figure 2.7. ZEUS Agent Architecture. From [Collis & Ndumu].

ZEUS is ideal for the GMD implementation because it (1) is suitable in a global system with distributed platforms to be used by different user groups, a wide variety of machine platforms, and different operating systems; (2) is JAVA based, which is the leading agent programming language; and (3) is highly suited for agent-to-agent communication [Bissel et al].

2. GMD Comparison Notes

The primary difference in this approach is that it does not use a layered agent structure and committee decision-making approach. Because of this, GMD does not have the same level of collaboration in decision-making and there is no voting among agents. Instead, the approach is based on the creation of agents as necessary to complete a task. Agents are created from the Registry by direction of the Agent Manager and can be destroyed when they are no longer needed.

H. ATR COMMUNICATIONS RESEARCH LABORATORIES

In this model, adaptive QoS management is achieved by direct and indirect cooperation of layered multi-agent system. The proposed QoS management platform can flexibly adapt to user's QoS requirements, kinds of systems, and large variations in system states, through QoS negotiation in the upper layer of the multi-agent system. It can quickly adapt to small fluctuations of system states through QoS adaptation in the lower layer of the multi-agent system. [Kosuga et al]

Figure 2.8 shows the model of the ATR adaptive QoS management platform. Figure 2.9 maps the relationships between the different levels of QoS.

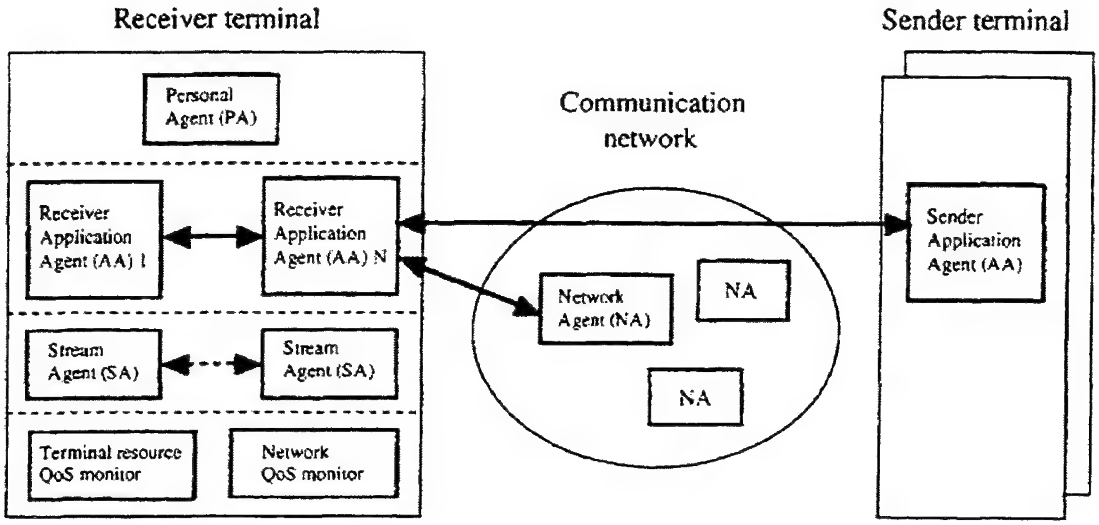


Figure 2.8. ATR Model of the Adaptive QoS Management Platform. From [Kosuga et al].

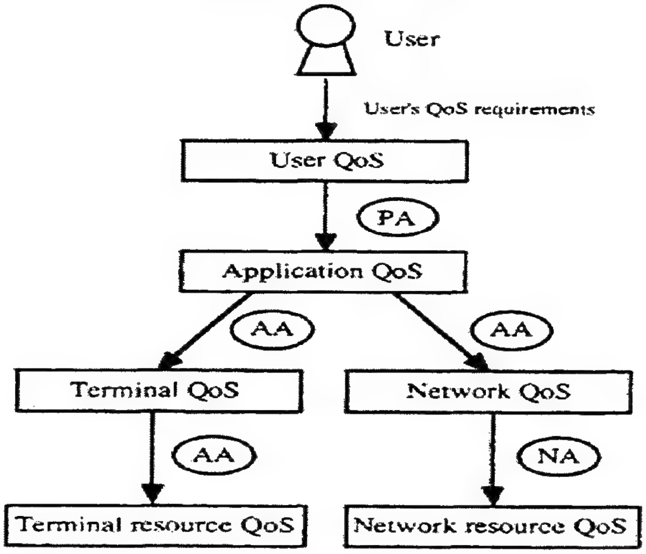


Figure 2.9. ATR QoS Levels and QoS mapping. From [Kosuga et al].

The *Personal Agent (PA)* supports a communication user, and one of its functions is mapping between the user QoS and the application QoS. The PA executes this QoS mapping by considering the user's character and preference, and generates a utility function that indicates a relationship between the application QoS and corresponding user's utility.

The upper layer of the MAS is composed of several *Application Agents (AA)* and *Network Agents (NA).* These agents are deliberative and perform QoS negotiation by directly exchanging messages. The lower layer of the MAS is composed of *Stream Agents (SA).* These agents are reactive, autonomous, and cooperative with each other indirectly through common memory.

One AA is created for a multimedia application. The AA also executes QoS mapping from the application QoS to the terminal resource QoS and the network QoS. The terminal resource QoS is allocated to receiver applications in a terminal through intra-terminal QoS negotiation in which the corresponding receiver AAs participate.

Many NAs are distributed in the communication network. Each NA locally manages network resources and executes QoS mapping from the network QoS to the network resource QoS. The network QoS is determined through QoS negotiation between the AA and the NAs. The inter-AA-NA QoS negotiation is performed based on the network QoS because the AA cannot manage the widespread network resources and only the network QoS can be monitored in the terminal.

One SA is created for each media stream. If a multi-media application handles several media streams, several SAs are created corresponding to one AA. Each SA performs QoS adaptation autonomously according to fluctuation of monitored terminal resource QoS and network QoS. Here, QoS adaptation means adjustment of the application QoS in pre-determined range, and can be realized by manipulating the flow control function. The range of QoS adaptation is given by the PA in advance. The application QoS is realized by best effort in this range. The PA also gives QoS adaptation policy, such as stream priority and QoS parameter priority, in advance.

When the fluctuation of system states is relatively large and the application QoS need be adjusted beyond the predetermined range, the SAs request QoS renegotiation in the AAs. For this reason, if the range of QoS adaptation is too narrow, the QoS negotiation will be initiated frequently and communications may often be disturbed. Necessity of QoS renegotiation can be regarded as a variation of the QoS mapping function.

1. ATR Comparison Notes

The ATR approach also uses a layered agent framework with increasing levels of decision-making and responsibility. The primary difference is the comparative lack of intra-agent collaboration within the layers. Decision-making is more vertical. As a second note, there is no refillable knowledge base such as the case based reasoning library. All the intelligence resides within the agents themselves.

I. RETSINA

RETSINA, developed at Carnegie Mellon University, stands for Reusable Environment for Task Structured Intelligent Network Agents. The RETSINA framework is composed of distributed collections of intelligent software agents that cooperate asynchronously to perform goal-directed information retrieval and information integration in support of performing a variety of decision-making tasks. A collection of RETSINA agents forms an open society of reusable agents that self-organize and cooperate in response to task requirements. Their designer focused on three crucial characteristics of the overall framework that differentiate RETSINA from others [Sycara et al]:

- Use of a multi-agent system where the agents operate asynchronously and collaborate with each other and their user(s)
- Agents actively seek out information
- Information gathering is seamlessly integrated with problem solving and decision support

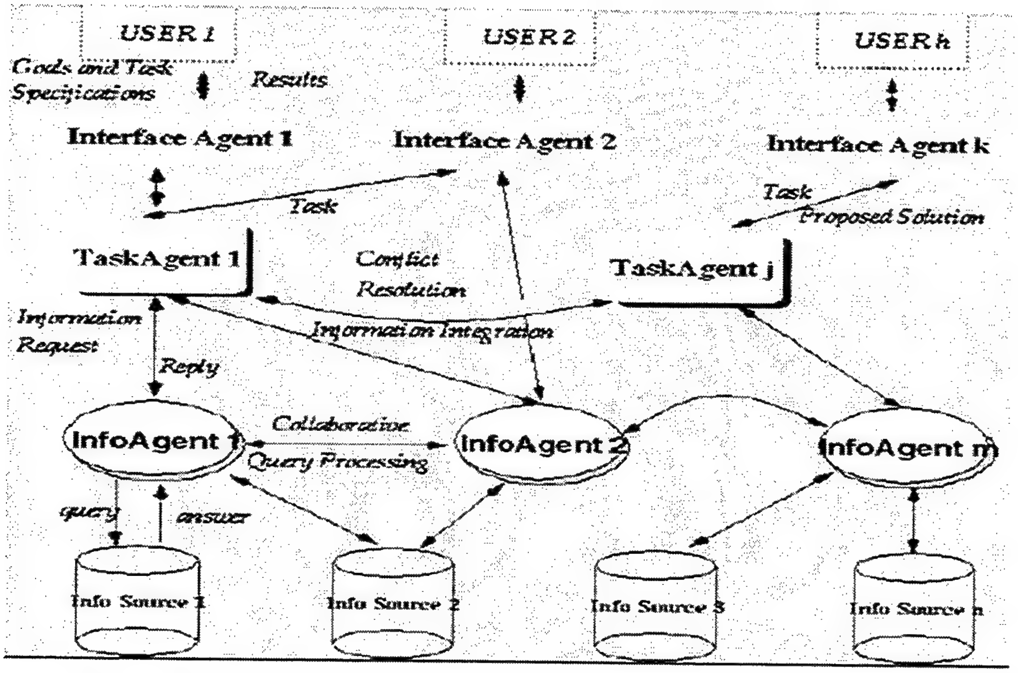


Figure 2.10. RETSINA Agent Organization. From [Sycara et al].

Figure 2.10 shows RETSINA's three types of agents: *interface, task*, and *information. Interface Agents* interact with the user by receiving user specifications and delivering results. The main functions of Interface Agents include: collecting relevant information from the user to initiate a task; presenting relevant information, including results and explanations; asking the user for additional information during problem solving; and asking for user confirmation when necessary. *Task agents* support decision making by formulating problem solving plans and carrying them out through querying and exchanging information with other agents. The Task Agent receives user delegated task specifications from an Interface Agent; interprets the specifications and extracts problem solving goals; forms plans to meet these goals; identifies information-seeking sub-goals present in its plans; and decomposes the plans and coordinates with appropriate task agents of information agents for plan execution, monitoring, and results composition. *Information Agents* provide intelligent access to a heterogeneous collection of information sources. They answer one-shot queries about associated information sources; answer periodic queries that will be run repeatedly and send results to the requestor each time; and monitor an information source for a change in a piece of information. [Sycara et al]

1. RETSINA Comparison Notes

RETSINA was originally designed for information brokerage in an open system such as the Internet. However, the MAS structure and basic agent principles are similar to the above approaches making it applicable to this research. Earlier applications included financial portfolio management, E-commerce, and logistics. Currently, the usage of RETSINA is being expanded into the area of network management. Based on its capabilities and past success, it is highly conceivable that RETSINA can feasibly be utilized for the purposes of adaptive QoS management. In this area, the Task Agent finds the best QoS match for the user based on the information available on the network. The primary difference in this framework is the lack of learning (it is optional) and corresponding lack of a knowledge base. Collaboration is accounted for but is not based on the same committee decision-making voting principle.

J. HYBRID

The Hybrid demonstrator addresses the performance and configuration management of semi-permanent Virtual Paths (VPs) of an ATM network. It uses techniques from intelligent agent research to provide support for distributing management tasks among a hierarchy of autonomous controllers. Each controller is a goal-directed agent with local control of a set of ATM resources. However, agents coordinate their activities to ensure system-wide and regional objectives are maintained through the exchange of goal requests and constraints on behavior. The demonstrator is an implementation of a distributed agent framework in CORBA and Java, and uses the KQML standard for agent communication. Specific agents have been developed in C++ and the expert system language CLIPS, which implement adaptive cost-based routing algorithms, trading protocols and encode service/business rules. [Somers et al]

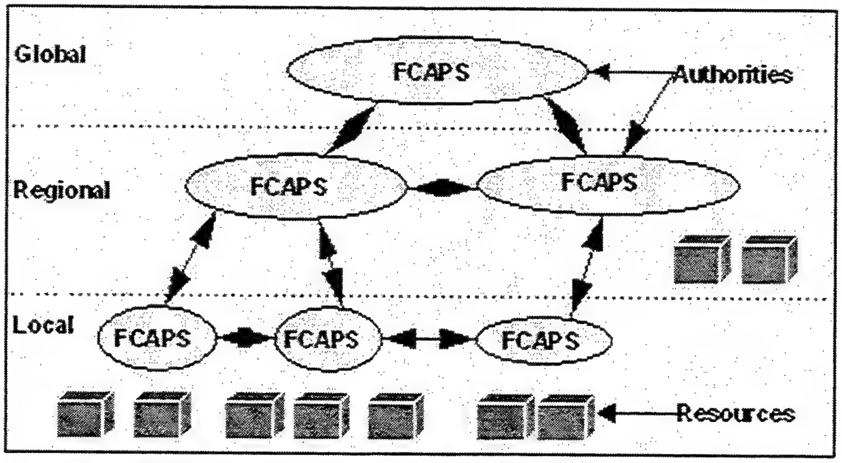


Figure 2.11. Hybrid Architecture. From [Somers et al].

The Hybrid system provides an approach to distributing the management responsibilities for an ATM network between a hierarchy of cooperating controllers, termed *authorities*, and identifies a number of specific conventions, which can be used to ensure the independent actions of authorities are coordinated to maintain system goals. Central to this approach is the deployment of goal-directed intelligent agents that are imbued with local problem-solving capabilities. Behavioral interaction through the exchange of goals, rather than parameterized function calls, insulates the activities of individual agents from each other at the computational level. The coordination conventions are necessary to coordinate the activities of individual agents at the task level. The conventions have been designed to minimize the amount of communication that is necessary between agents.

1. Hybrid Comparison Notes

Hybrid is the predecessor to Tele-MACS and IMPACT. All three of these programs are based on the notion of Virtual Paths (VPs) for assured access. Similar to Tele-MACS, it is set up hierarchically in three layers, i.e., local (layer 1), regional (layer 2), and national (layer 3) as shown in Figure 2.11. Each of the layers is responsible for a particular region of the network, wherein various authorities negotiate resource availability. The primary difference with this project and Tele-MACS is that Tele-MACS agents are in control of more dynamic resources and are not tied to a static region of the network. In comparison to our proposed framework, the primary differences are structure, VPs, and learning.

K. TELECOMMUNICATIONS MULTI-AGENT CONTROL SYSTEM (TELE-MACS)

Tele-MACS is a multi-agent based system that manages the logical configuration of resources in an ATM type network. The system consists of multiple interacting agents, which have various roles to play in organizing the configuration problem. There are two main layers of control: a *planning layer* consisting of multiple planning type agents, and a *reactive layer* consisting of relatively simple agents that have time constrained tasks to conduct. The system makes sure that connections (calls) are placed onto the network in an organized manner such that the network configuration can be reorganized periodically. The planning metric used is derived from the need to maintain network survivability, so as to prevent a single point of failure. [Hayzelden et al].

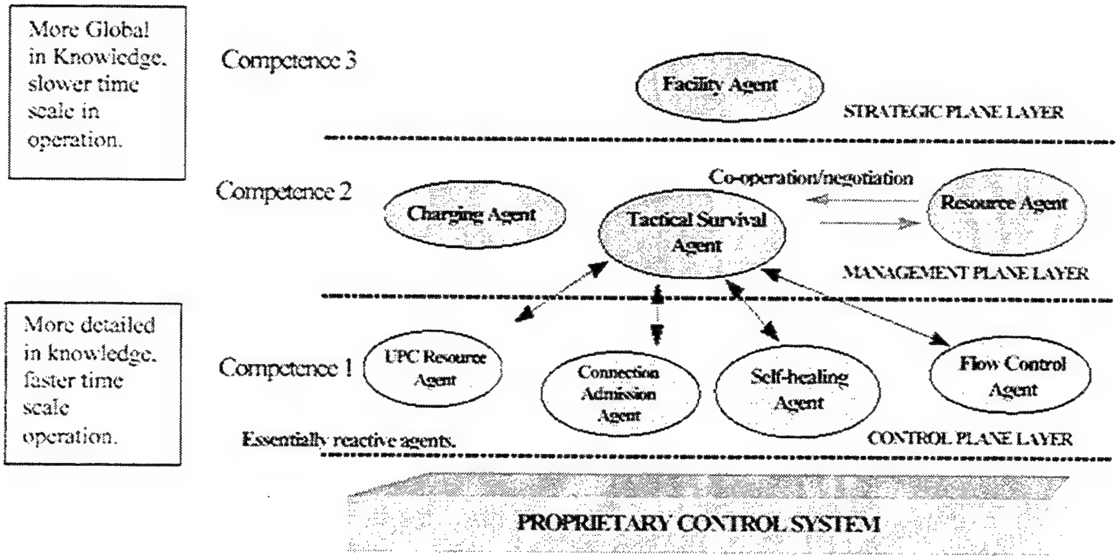


Figure 2.12. Tele-MACS Architecture. From [Hayzelden et al].

Figure 2.12 shows the multi-layer multi-agent control architecture. The main idea behind the Tele-MACS approach is the building of the complete coordinated MAS system that achieves the specified purpose to a certain degree of competence. The system is tested and adjustments are made until the system layer (consisting of interacting software agents) satisfactorily meets its intended purpose. Next, another complete layer of control is built to a higher level of competence and coupled through a suppression mechanism. The key elements to note in using this approach are that:

- Layering allows the isolation or encapsulation of interacting agents within a certain environment (well-defined interfaces between the layers are created).
- Provides *robustness* to software failure and robustness against the inability to reach an action sequence within the time interval of the control loop (the layers can operate at different time scales).

All of the agents in the system are autonomous entities that conduct activities to solve the bandwidth configuration problem. *Control plane agents* carry out operations considering 'emulated views of the world'. Control plane agents are relatively simple reactive agents that conduct actions based on some default rules. These rules can be overruled or 'tricked' when a more competent agent (an agent in a higher competence layer) requires a control plane agent to conduct a different action sequence, such that its goal can be achieved.

The *Management Agents* are based on the deliberative agent paradigm. As such, they have a greater global awareness of the network's state and operate at a slower time scale for action activation. They are therefore given the opportunity to generate planned solutions to the problem. When a plan is generated they influence the actions of the reactive control plane agents by passing them an emulated view of the world (this is a suppression signal or message that alters the beliefs of the agent). The emulated view of the world invokes changes to the reactive agents' actions.

The *Facility Agent* operates on the strategic management layer. It is a higher-level planner that is only invoked when the logical topology cannot deal with the demand. It therefore, generates plans for alterations to the physical topology (this planner is not fully implemented in the current system).

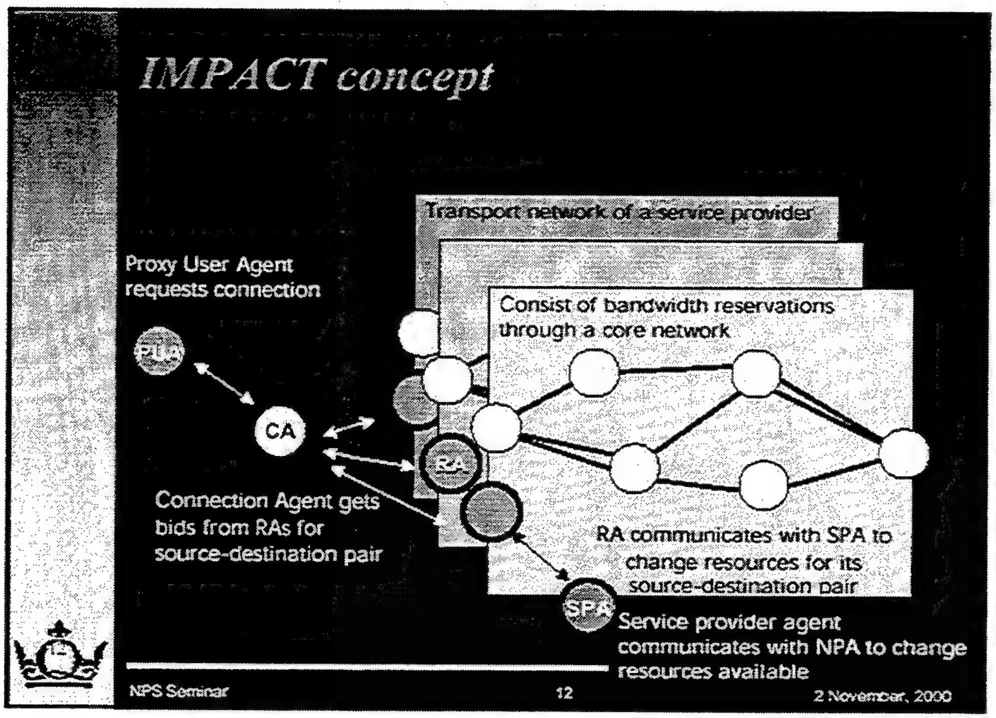
1. Tele-MACS Comparison Notes

Tele-MACS uses a tri-layer approach for agent decision-making, where each layer is defined to conduct control of the network infrastructure to a certain level of competence. This approach is an ideal fit for solving the problems of wide distribution and robustness. The unique feature with Tele-MACS is that it subsumes the proprietary control system, whereby it functions in its intended fashion. Tele-MACS merely adds intelligent control. While the principle of hierarchy is similarly used in our proposed framework, there are differences in QoS layer, structure, VPs, level of collaboration, and learning.

L. ACTS PROJECT: IMPACT

The IMPACT project represents another type of management system for ATM networks that uses concepts from the multi-agent systems paradigm. The intelligent multi-agent system has been applied to improve upon conventional ATM connection admission procedures (CAC) by using the cooperative planning abilities that the agent paradigm allows. The introduction of cooperative abilities leads to enhancements in terms of allowing more negotiable resource allocation management procedures. [Bigham et al]

Communication between end users to set up a connection usually involves signaling, which is responsible for the conventional step-by-step routing and CAC. It is assumed that the network resources will be managed by exploiting dynamic bandwidth allocation to virtual paths (VPs), which is defined as a path of specified bandwidth from a source node in the network to the destination node in the network, using physical links of the network. Only source-to-destination VPs are considered in the resource management model, i.e. no path segments. No routing is done for individual virtual connections. Instead, all new connections are allocated to one of the relevant VPs. The bandwidth associated with any VP can change continually and is one of the controllable parameters for the Network Service Provider (NSP) or negotiation commodity for Service Providers (SP) who is not the Network Provider (NP). IMPACT believes this to be a highly realistic assumption for the management of a complex network. It is assumed that the set of VPs associated with a source-destination pair is known, fixed in terms of route (though not bandwidth), and is a small manageable subset of the set of possible VPs for that source-destination pair. While this sounds limiting, IMPACT does not believe this to be so in practice as the set of enumerated VPs could be changed over time. Pre-enumeration of the VPs simplifies the CAC mechanism. [Bigham et al]



IMPACT concept

Transport network of a service provider

Proxy User Agent requests connection

Consist of bandwidth reservations through a core network

Connection Agent gets bids from RAs for source-destination pair

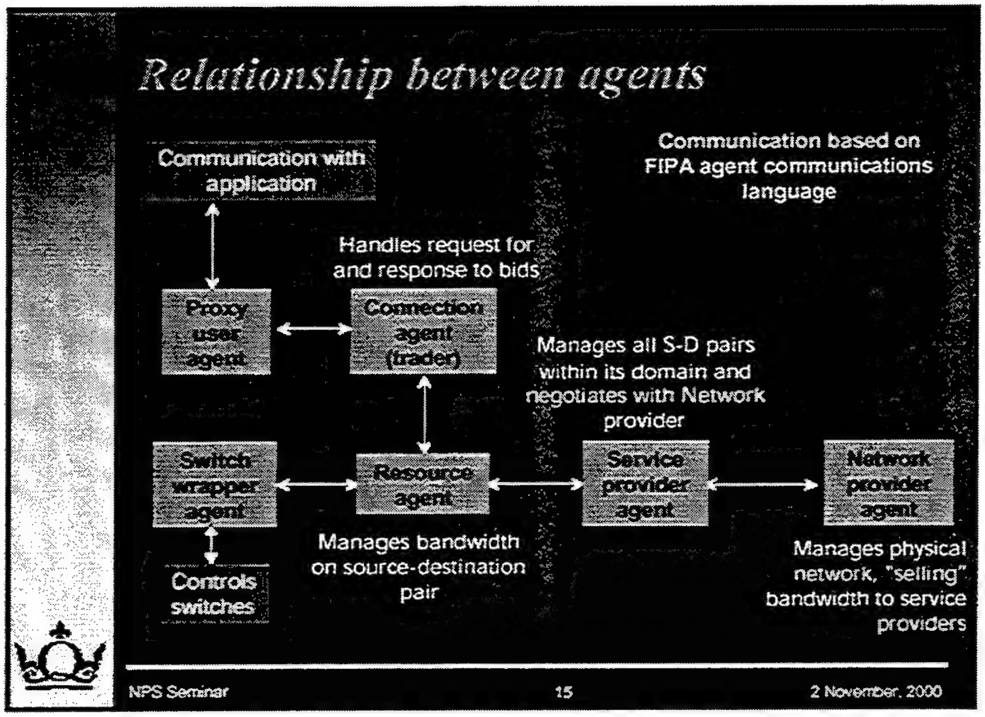
RA communicates with SPA to change resources for its source-destination pair

Service provider agent communicates with NPA to change resources available

NPS Seminar 12 2 November, 2000

Figure 2.13. IMPACT Concept. From [Cuthbert].

Resource Agents (RA) (Figure 2.13) manage the VP connections based on costs and feasibility of connections. In the model (see Figure 2) each RA of a SP manages the resources of a pre-defined set of VPs from a source to a destination. The bandwidth of each VP is dynamic, in the sense that it is subject to adjustment, by negotiation, with the NP. For the case of the NSP, this could reduce to simple compliance with the wishes of the NP. There is an RA for every source-destination pair and each RA contains sub-agents for each traffic class.



Relationship between agents

Communication with application

Communication based on FIPA agent communications language

Handles request for and response to bids

Proxy user agent

Connection agent (trader)

Manages all S-D pairs within its domain and negotiates with Network provider

Switch wrapper agent

Resource agent

Service provider agent

Network provider agent

Controls switches

Manages bandwidth on source-destination pair

Manages physical network, "selling" bandwidth to service providers

NPS Seminar 15 2 November, 2000

Figure 2.14. IMPACT Agent Relationships. From [Cuthbert].

The main agents in the system are the CAC Agents (CACAs), the Resource Agents (RAs), the Proxy User Agents (PUAs), the Proxy Connection Agents (PCAs) and the Switch Wrapper Agents (SwWrAs). These are shown in Figure 2.16. Just as in the Tele-MACS project, the IMPACT project also utilizes reactive and planning layers for agent decision-making.

1. IMPACT Comparison Notes

The IMPACT project represents an interesting approach to adaptive QoS management if applied to military C4ISR networks. While money is clearly not how service arrangements would be made in tactical military environments, the notion of negotiating for services based on priority and service level agreements has parallel applicability to military C4ISR. Under this system, Resource Agents negotiate for the best services for the user via virtual paths. It is this concept of virtual paths that is the primary difference with our proposed agent framework. Other differences include agent types and agent placement.

M. SUMMARY

In closing, this chapter discussed the multi-agent system paradigm as a means for adaptive QoS management in a dynamic environment. We introduced our proposed agent framework based on collaboration, adaptability, case based reasoning, the committee decision model, and a layered artificial neural network decision-making framework. Subsequently, we highlighted various agent architectures that might also satisfy the task, but in a different manner. The similarities/differences are summarized in Tables 2.1/2.2.

Agent Framework	**Description**	**Agent Types**	**Language**	**Agent Structure**
Proposed	Translates service level requirements	Router, Bridge, Agents-Facilitators	Java	Layered; ANN
Proteus	Network Management/ Variable Polling	Task, Performance Trending, MIB Server, Polling	Java	Uses task agents
GMD	Service Level Management	Interface, Agent Manager, Task	Java	Agent Manager; Registry
ATR	Distributed multi-media applications	Personal, Application Stream Terminal Resource, Network Resource, Network	Java	Layered
RETSINA	Information brokerage	Interface, Task, Information	Java, C, C++, Python, List, Pearl	Open system; Heterogeneous agent types
Hybrid	Network management; Virtual Paths	Service, Proxy, Resource, Performance, Configuration	Corba, Java, C++, CLIPS	Layered; Regional
Tele-MACS	Source Destination Virtual Paths; Based on SLAs; Telecommunications	Facility, Resource, Tactical Survival, Charging, UPC Resource, CAC, Self-Healing, Flow Control	Java	Hierarchical: reactive and planning; Levels of competence
IMPACT	Virtual Paths; Highest bidder; Based on SLAs; Telecommunications	CAC, Resource, Proxy User, Proxy Connection, Switch Wrapper	Java	Layered: reactive and planning

Table 2.1. Comparison of Agent Approaches.

Agent Framework	**Learning**	**Collaboration Levels**	**Toolkit**	**IP/ATM**	**Comments**
Proposed	CBR	Vertical/ Horizontal		Both	Best combination of capabilities
Proteus	Temporal Difference	None		ATM	Not currently adapted for meeting service level requirements
GMD	Yes	Vertical/ Horizontal	ZEUS	IP	Agents are created/killed as necessary
ATR	Yes	Vertical		ATM	Does not have horizontal collaboration
RETSINA	Optional	Vertical/ Horizontal		IP	Could be developed for service level management
Hybrid	Yes	Vertical/ Horizontal		ATM	Not designed for service layer management
Tele-MACS	Yes	Vertical/ Horizontal	BAT	ATM	Telecommunications applications; Several concepts used in IMPACT
IMPACT	Yes	Vertical/ Horizontal	BAT	ATM/ (Both)	More media than Tele-MACS; Must convert concept of highest bidder to one based on military vice monetary priority

Table 2.2. Comparison of Agent Approaches (Continued).

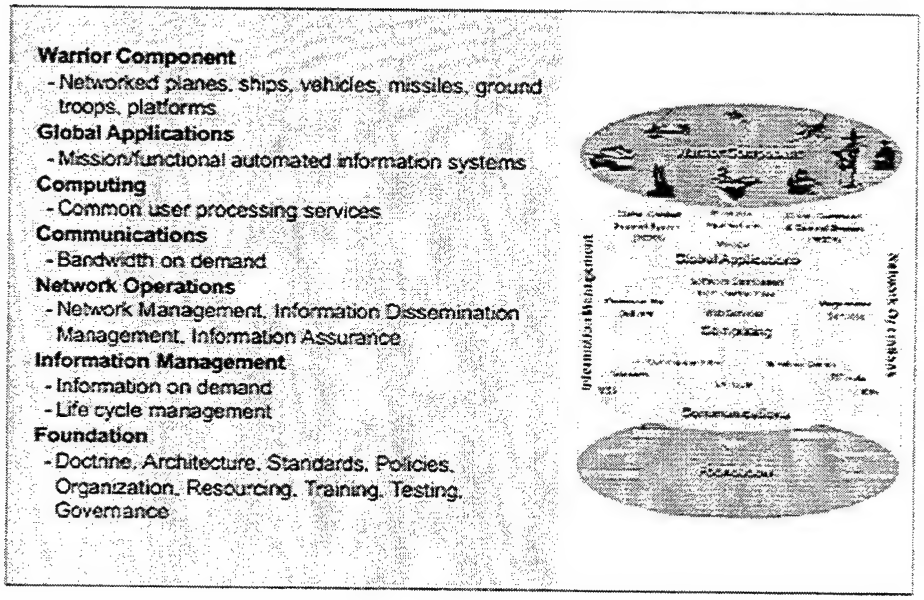
III. ADAPTIVE QOS MANAGEMENT

In this chapter, we investigate the underlying concepts behind the successful implementation of multiple collaborative agents for adaptive QoS management and build on our chosen agent framework from Chapter II. These fundamentals include Service Level Management (SLM), the Telecommunications Management Network (TMN) framework, Quality of Service (QoS), and Policy Based Management (PBM). In accordance with user profiles and policies, intelligent agents adapt to a dynamic environment by utilizing network resources and channels to optimally translate the user's desires across the network. The agents bridge the interface between the user's service level requirements and the network management requirements in the TMN framework. Utilizing important concepts from SLM, PBM, and TMN are critical in understanding and developing this capability.

We follow a systems level analysis methodology to develop SLM techniques in capturing application requirements between users and service providers. In the next chapter, we apply these techniques to acquire real-world application requirements for an actual C4ISR network at the Pacific Region Network Operating Center (PRNOC) in Wahiawa, Hawaii.

A. INTRODUCTION

In the forthcoming age of Network Centric Warfare, there exists a strong need to be able to sift through the multitude of information in order to attain information superiority and thereby prevail over the enemy. For the warfighter, information is of no value if it cannot be used in time to have an impact on his decisions. This is the crux of information superiority, whereby we can get inside the enemy's so-called OODA loop [Barneyback]. For the warfighter operating on the warrior component of the Global Information Grid (GIG), information superiority means not only having the latest information made possible by the latest advances in information technology, but also having the ability to effectively decipher and use it to have a decided advantage over the enemy.



Warrior Component
- Networked planes, ships, vehicles, missiles, ground troops, platforms

Global Applications
- Mission/functional automated information systems

Computing
- Common user processing services

Communications
- Bandwidth on demand

Network Operations
- Network Management, Information Dissemination Management, Information Assurance

Information Management
- Information on demand
- Life cycle management

Foundation
- Doctrine, Architecture, Standards, Policies, Organization, Resourcing, Training, Testing, Governance

Figure 3.1. Global Information Grid. From [JV2020].

In parallel fashion, the management of C4ISR networks has its own bearing on information superiority, albeit in a different way as shown in Figure 3.1. Network management is integral to the Global Information Grid (GIG) in that it maximizes its usability as a whole and permeates through all levels. In this capacity, the benefits of network management of C4ISR systems are somewhat less apparent because of its behind-the-scenes nature.

In truth, the goal of good network management is to be as transparent to the warfighter as possible because when this is true, all application requirements are being met. On the other hand, bad network management disrupts the decision cycle of the warfighter before he even gets a chance to act on the information, regardless of his command and control capabilities. The underlying reality of network QoS management is that it only really becomes recognizable to the warfighter when it does *not* deliver the user's desired application requirements. Intelligent agents can help make this effort more efficient and seamless.

B. SERVICE LEVEL MANAGEMENT

Multiple collaborative agents are ideally suited to match the service level requirements of the warfighter in a transparent manner. Service Level Management (SLM) refers to:

> The process of negotiation, service level agreement (SLA) articulation, checks and balances, and reviews between the supplier (NOC) and consumer (warfighter) with respect to the services and service levels that support the consumer's business practices [Lewis 1998, p.2].

In other words, SLM provides a formal method for optimizing the C4ISR network; that is, by best meshing the desires of the warfighter with the capabilities of the network service provider (NOC).

Today, SLM is a buzzword in the IT industry. Leading software vendors and service providers all claim SLM support because it provides much more than just network management. The identified key benefits of SLM are [Bissel et al]:

- QoS measurement

- Definition of required performance
- Alignment of information technology with business
- Setting/management of expectation

Learning and understanding the needs of the user (warfighter) is the first step in SLM, and is not necessarily as easy as it can seem. The warfighter and network manager have a different language when discussing requirements. Moreover, the two camps have different perspectives in how to map the well-being of elements in the infrastructure into the well-being of the services. The differences can be summarized as follows:

- Parameters that are easy to understand and measure for network specialists do not translate well into parameters that are easily understood by ordinary customers.
- Parameters that are easily understood by customers are not easy for network specialists to measure.

This disparity is known as the "Semantic Disparity Problem" and overcoming it is generally recognized as the crux of SLM. [Lewis 2000]

C. GATHERING REQUIREMENTS

To develop application requirements, we follow a systems level analysis methodology because it coincides with SLM and provides a good starting basis with which to communicate with both the user and service provider. By understanding the network in terms of levels, we can better distinguish the specific QoS needs of the user and understand the inter-relationships of the various network components with respect to QoS. Requirements add to each other, such that application requirements add to user requirements, host requirements add to application requirements, and all add to network requirements. As a result, requirements filter down from user to application to host, resulting in a service request that is a set of service requirements, or service levels, to the network that correspond to different levels of the TMN layer architecture. This results in a service offering that is end-to-end, consisting of service requirements that are configured in each element (e.g., router, bridge, circuit, etc). [McCabe]

The network analysis process begins with requirements analysis, which is about understanding the design environment. This process consists of: (1) identifying, gathering, and understanding system requirements and their characteristics; (2) developing thresholds for performance to distinguish between the low and high performance services; and (3) determining specified services for the network [McCabe]. Understanding application requirements is a necessary first step in order to effectively program the agents.

1. Semantic Disparity Problem

The first step in network analysis is to communicate with the customer to understand his needs. In SLM, there are basically three different approaches to providing service and overcoming the so-called Semantic Disparity Problem. These include the [Lewis 2000]:

- *User-Centric Approach,* whereby service providers find some way to measure the parameters of interest to customers.

- *Happy-Medium Approach,* whereby the service provider and user search for service parameters that are easy to measure and meaningful to the user at the same time.

- *Techno-Centric Approach,* whereby service providers show the users how low-level network, systems, and application parameters translate into higher-level parameters that reflect the health of the consumers' services.

Obviously, the user is most important, but the user centric approach is not always the most feasible. In the final analysis, the prime parameter of interest is simple user happiness, which is hard to measure in any case.

2. Standard User Requirements

From the model of a generic system, the user (warfighter) component is at the highest layer. From the warfighter's perspective, his main concern is getting the system to meet his application requirements. Thus, the system should be able to adapt to the warfighter's environment, provide quick and reliable information access and transfer, and offer quality services to the user. At the highest level, standard requirements are generally classified in terms of the following [McCabe]:

- *Timeliness:* User is able to access, transfer, or modify information within a tolerable time frame.

- *Interactivity*: A measure of the response time of the system when it is required to actively interact with a human.

- *Reliability*: A requirement for consistently available service.

- *Quality*: Refers to the quality of the presentation to the user.

- *Adaptability*: Ability of the system to adapt to the users' changing needs.

- *Security:* A requirement to guarantee the integrity (accuracy and authenticity) of the user's information and physical resources, as well as access to the user's and system's resources.

- *Affordability:* The cost of obtaining these services must be within a reasonable price range.

These user requirements form the basis for performance requirements of the network. From the performance requirements, applications can be grouped into general classifications.

3. Application Requirements

Services in the network can be described by the performance requirements *reliability, capacity,* and *delay,* which form the basis of service layer QoS requirements from the TMN architecture. *Reliability* (determinism/accuracy) is a measure of the system's ability to provide deterministic and accurate delivery of information. *Capacity* (bandwidth/throughput) is a measure of the system's ability to transfer information. *Delay* (latency) is a measure of the time differences in the transfer and processing of information. [McCabe]

In general, applications were primarily designed to support basic connectivity and data transfer between hosts, but the user and network requirements have started to play an ever-increasing role. For this reason, deriving an understanding of requirements is critically important to network management. In doing so, we can derive general application classifications in terms of priority as listed below [McCabe]:

- *Mission critical applications* that have specified (deterministic and/or guaranteed) reliability
- *Controlled-rate applications* that have specified capacity
- *Real-time* (and possibly *interactive) applications* that have specified delay

Different applications have different reliability, capacity, and delay (i.e. QoS) requirements for general applications. These are discussed in greater detail in the next section.

D. QUALITY OF SERVICE

The challenge of network management is to consistently deliver high levels of performance. This has become increasingly difficult due to higher bandwidth requirements for applications and the unpredictable nature of application deployment. As a result, QoS can fluctuate from day to day. At PRNOC, this can be due to ships deploying, contingency operations, or system degradation.

In broad terms, the QoS of a wide area network (WAN) is a measure of how well it does its job, i.e., how quickly and reliably it transfers various kinds of data from source to destination. With the growth of packet switching and the spread of many kinds of communications traffic, there is more than one set of criteria to satisfy. For example, the data rate needed for satisfactory voice communication may take an intolerable time to transfer high-resolution images. On the other hand, the degree of network latency acceptable in transferring some files may not be adequate for real-time voice.

Technically, QoS refers to an aggregation of system performance networks. According to most literature, the following are usually recognized as highly important:

1. Availability

Ideally, a network is available 100 percent of the time, but this is obviously not always the case. Even so high a figure as 99.8 percent translates to about one and half down hours per month [Dutta-Roy].

2. Throughput

Throughput is the effective data transfer rate measured in bits per second. Throughput is not synonymous with bandwidth, which is merely the size of the pipe. In contrast, throughput takes into account such factors as number of users, bit overhead for identification or other purposes, and line degradation.

3. Packet loss

Network devices, such as switches or buffers, sometimes have to hold data packets in buffered queues due to congestion. If the link remains congested for too long, the buffered queues overflow resulting in packet loss. In turn, the lost packets must be re-transmitted resulting in a longer total transmission time. [Dutta-Roy]

4. Latency

Latency is delay introduced in application traffic flowing across a network path due to queuing, processing, or congestion. Other sources of delay include propagation, transmission, routing, and satellite propagation. For the public Internet, a voice call may easily exceed 150 ms of latency due to signal processing or congestion [Dutta-Roy]. From an application service perspective, optimizing the total end-to-end delay is more important than individual sources of delay.

5. Jitter

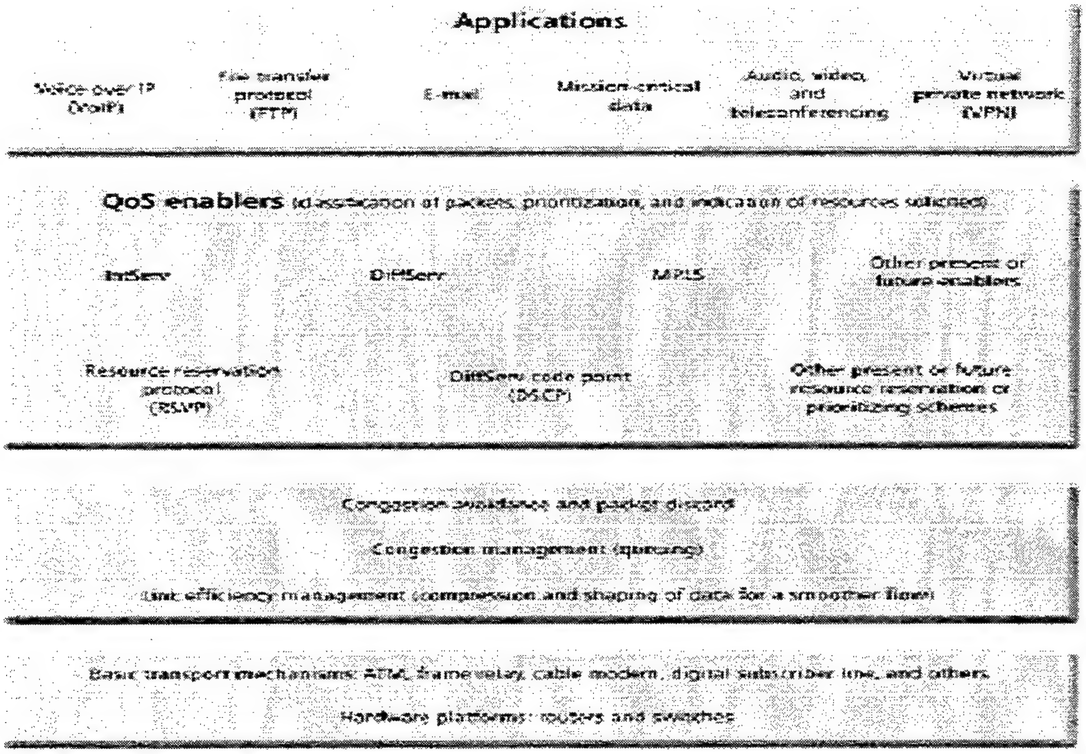
Jitter is the distortion of the inter-packet arrival times compared to the inter-packet times of the original transmission (i.e. delay variance). Causes include variations in queue length, variations in the processing time needed to reorder packets that arrived out of order due to different paths, and variations in the processing time needed to reassemble packets that were segmented by the source before being transmitted [Dutta-Roy]. Jitter is particularly demanding to multi-media applications.

Application Types	**QoS Requirements**			
	Bandwidth	**Latency**	**Jitter**	**Packet Loss**
ERP Applications	Moderate	Low	-	Low
Legacy SNA Applications	Low	Low	-	Low
Productivity Applications	Low/Moderate	Moderate	-	-
E-mail	Low	-	-	-
File Transfer	Bursty High	-	-	-
Thin Clients	Low to Moderate	Low	-	Low
Video-conferencing	Sustained High	Low	Low	Low
Voice over IP	Sustained Moderate	Low	Low	Low
Streaming Media	Sustained Moderate to High	Low	Low	Low
Server Load Balancing	QoS requirements are application and server dependent			

Table 3.1. QoS Requirements. After [Extreme].

Applications differ in the way they use bandwidth and their QoS requirements (Table 3.1). The unpredictable mix of applications running on a dynamic network and the conflicts that occur due to simultaneous application requirements induces QoS problems. This is the fundamental dilemma for QoS resource management and the driving impetus behind using intelligent agents. “Throwing bandwidth at the problem” is not sufficient in itself to guarantee that specific applications will perform adequately under all traffic conditions. The bandwidth must be intelligently managed to prioritize application requirements and business priorities.

The various enabling methods for QoS on the network management layer of the TMN functionality are shown in Figure 3.2. Data from one or more applications [top] flow down through QoS enablers that, in turn, prioritize the data flows and indicate the resources each requires. The data then continues through various levels of software and hardware that control packet discard mechanisms on the next level when buffered queues become too long. Finally, the data reaches the basic transport mechanisms and their hardware platforms that carry packets to the next node. [Dutta-Roy]


Applications
Voice over IP (VoIP)
File transfer protocol (FTP)
E-mail
Mission-critical data
Audio, video, and teleconferencing
Virtual private network (VPN)
QoS enablers (classification of packets, prioritization, and indication of resources solicited)
IntServ
DiffServ
MPLS
Other present or future enablers
Resource reservation protocol (RSVP)
DiffServ code point (DSCP)
Other present or future resource reservation or prioritizing schemes
Congestion avoidance and packet discard
Congestion management (queuing)
Link efficiency management (compression and shaping of data for a smoother flow)
Basic transport mechanisms: ATM, frame relay, cable modem, digital subscriber line, and others
Hardware platforms: routers and switches

Source: Cisco Systems Inc.

Figure 3.2. QoS Application Requirements. From [Dutta-Roy].

Note that the above factors are only discussed to illustrate the abstraction of service level requirements to network management requirements; the primary concentration of this work remains the higher service level requirements.

E. POLICY BASED MANAGEMENT

Meeting QoS requirements under dynamic conditions can be tied to Policy Based Management (PBM). PBM is defined as "the combination of rules and services where rules define the criteria for resource access and usage" [Vicente et al, p. 2]. Instead of getting involved in the details of queuing mechanisms and configuring routers and switches, PBM allows the network manager to simply define a policy that might say, "give my SAP application guaranteed bandwidth and the highest priority." PBM simplifies the details of policy implementation and operates as shown in Figure 3.3.

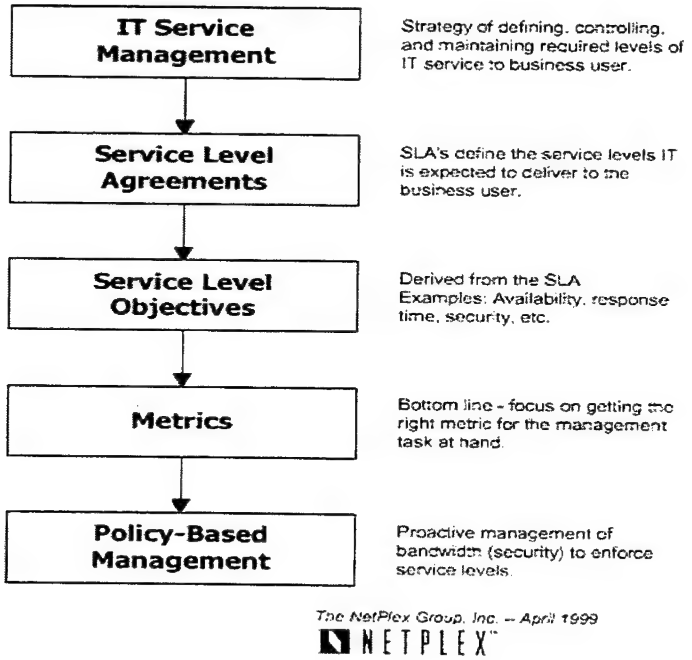

IT Service Management
Strategy of defining, controlling, and maintaining required levels of IT service to business user.
Service Level Agreements
SLA's define the service levels IT is expected to deliver to the business user.
Service Level Objectives
Derived from the SLA Examples: Availability, response time, security, etc.
Metrics
Bottom line - focus on getting the right metric for the management task at hand.
Policy-Based Management
Proactive management of bandwidth (security) to enforce service levels.
The NetPlex Group, Inc. – April 1999
NETPLEX

Figure 3.3. Policy Based Management. From [Hewlett-Packard].

As in SLM, PBM is accomplished via the Service Level Agreement (SLA). Ideal in concept, but difficult in reality, SLAs help the service provider and user to work together to establish specific expectations. The SLAs help translate the service layer requirements into the network management layer requirements, i.e. meet the SLM paradigm.

1. General Technology Requirements

PBM requires an in-depth recognition of general technology requirements. By understanding and accepting these requirements, it is possible to maximize the combined benefits of QoS and SLM. These requirements are summarized below [Vicente et al]:

a. Service Differentiation

This is the ability to manage the quality of the service or service delivery mechanisms in order to meet some predefined network based delivery/metrics. Enablers including IntServ, DiffServ, RSVP, etc., operate at the network management layer as shown in Figure 3.3.

b. Network Provisioning and Bandwidth Management

This is the ability to provide proactive bandwidth management by facilitating control and allocation of bandwidth through device configuration management. This becomes the primary focus of agent technology for this work. The network must be capable of facilitating multi-device network configuration and performing admission control or traffic segmentation.

c. Integration with Network Management Systems and Legacy Devices

The SLA must be able to co-exist with existing operational models, security requirements, and business computing models. Obviously, the technology must be able to support or address legacy system limitations that are especially common in DoD.

d. Scalability

The system must be designed from the beginning to be able to match growing needs.

e. Industry Standardization

Network technology should conform to industry standards and use best practices to support network device and policy management interoperability. With respect to this research, this obviously also includes agent technology.

f. Security

The technology should facilitate resource access control and authorization, and should provide integration support for authentication and accounting. While security is a major issue with agents, it is not covered in this work.

F. TELECOMMUNICATIONS MANAGEMENT NETWORK (TMN)

First introduced in the mid-1980's, TMN has become the globally accepted framework for the management of telecommunications networks. For the most part, it is described in the International Telecommunications Union – Telecommunication Standardization Sector (ITU-T) and other standards (Figure 3.4). The functional architecture of TMN is termed the logical layered architecture. It essentially categorizes the OSI management functionality into the following layers [Sidor]:

- *Business management layer*: concerned with managing from an enterprise perspective, including finance, budgeting, goal setting, and product and human resource planning.

- *Service management layer*: concerned with managing services to end customers or to other service providers, including handling service orders, complaints, and billing, and measuring the quality of services (QoS).

- *Network management layer*: concerned with managing the network from end-to-end, that is, of all network elements (NE) and interconnecting links as a whole; also provides support of all services.

- *Element management layer*: concerned with managing a subset of NEs, individually or collectively as a subnetwork; also includes functionality mediating between NE's and the remainder of the TMN.

The use of the term layer recognizes an implicit support hierarchy among the functionality. However, the architecture does not allow communications between non-adjacent layers. Higher-level layers are viewed as having a higher level of information abstraction compared to lower layers.

> From this perspective, network management functionality is viewed as more vendor-independent than element management, while service management functionality is viewed as more technology independent than network management [Sidor, p. 59].

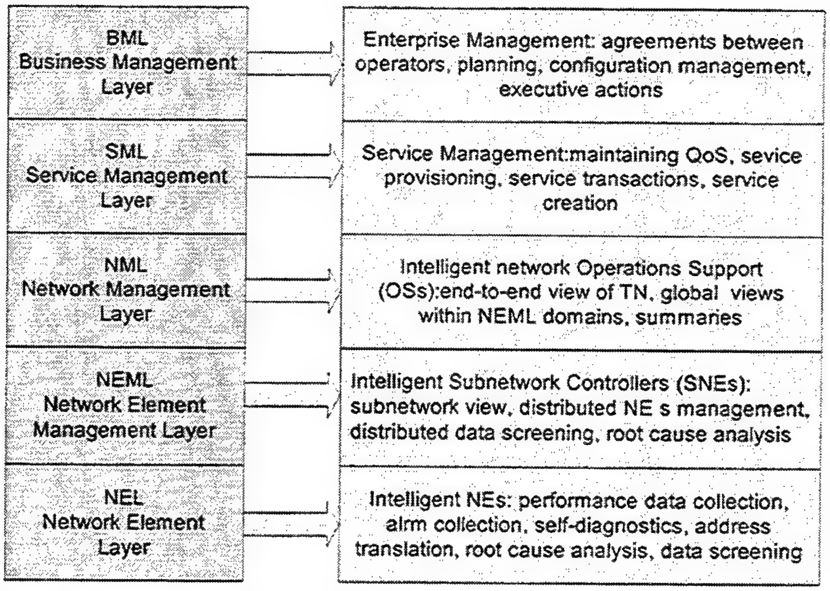

BML
Business Management Layer
Enterprise Management: agreements between operators, planning, configuration management, executive actions
SML
Service Management Layer
Service Management:maintaining QoS, sevice provisioning, service transactions, service creation
NML
Network Management Layer
Intelligent network Operations Support (OSs):end-to-end view of TN, global views within NEML domains, summaries
NEML
Network Element Management Layer
Intelligent Subnetwork Controllers (SNEs): subnetwork view, distributed NE s management, distributed data screening, root cause analysis
NEL
Network Element Layer
Intelligent NEs: performance data collection, alrm collection, self-diagnostics, address translation, root cause analysis, data screening

Figure 3.4. Telecommunications Management Network. From [Bordetsky].

G. DEVELOPING THE PROPOSED AGENT FRAMEWORK

For the remainder of this chapter, we build on the underlying concepts discussed above to further develop the proposed agent framework from Chapter II for the purpose of adaptive QoS management. First, we tie the TMN/SLM functionality to the fundamental concept of system coordination to address the problems of agent adaptation. Doing so allows the identification of critical relationships through associated feedback controls. From this perspective, the process of adaptive control and coordination in a multi-agent architecture can be based on the idea of mapping feedback control relationships into an agent's shared awareness memory, where feedback controls are delivered via agents-facilitators. In turn, this functionality is expanded into the agents' integration with case memory. [Bordetsky]

Unfortunately, real-time applications such as audio/video conferencing and shared application control have strict requirements in terms of delay and bandwidth as discussed earlier in the chapter. While asynchronous applications need only to adapt naturally via changes in response time, real-time applications must reduce the quality of the data stream to meet reduced bandwidth needs. To add to this, when multiple applications run simultaneously, lower-priority applications may be required to adapt to lower bandwidth usage or even be switched off entirely to free up bandwidth for higher priority applications.

1. Layers of Feedback Control: Individual Agent Adaptation

Two layers of feedback control *Call Preparation Control (CPC)* and *Connection Control (CC)* can be considered to support multiple applications. *Call Preparation Control* integrates feedback gathered from previous conferencing sessions to make informed decisions regarding connection setup and bandwidth tradeoff in future sessions. Its adaptation is long-term and mainly associated with the allocation of resources for the entire length of a multimedia call. *Connection Control* reflects ongoing performance measurement and adaptation throughout the length of the call. Its adaptation is short-term, such as may be required during a single call. The requirements of both layers of feedback control are summarized below [Bordetsky]:

a. Call Preparation Control Requirements

- A call must establish, modify, and execute voice, video, and multi-media application sharing communication between multiple users.
- A call must involve coordination between parties to satisfy response time, bandwidth, and other QoS requirements.

- A call contains relationships between user profiles, media, and system resources that may be dynamically modified during a call.
- Each user can request resources individually.
- A call will allow negotiations between different sites for system resources.

b. Connection Control Requirements

- Provided QoS parameters must be supervised.
- Flow control, congestion control, routing, reservation, and re-negotiation of resources must be provided for.
- Connections are modified and released.

2. Call Preparation Adaptation: Service Layer Feedback Controls

The proposed agent architecture can now be fully represented by the following components: (1) CBR memory, (2) agents-facilitators, and (3) collaborative feedback controls (Figure 3.5). The layers of case memory are structured according to the following feedback control relationship:

$$\text{SLM event } (t) = \{\mathbf{U(t)}, \mathbf{X(t)}, \mathbf{P(t)}, \mathbf{I(t)}\},$$

where:

SLM event = Service Level Management event; **U(t)** is a set of *user input controls* (desktop conference calls, links to knowledge sources); **X(t)** is a set of *SLM process state variables* (QoS restraints such as response time and bandwidth); **P(t)** is a set of *service process outputs*; and **I(t)** describes the *environmental impact* to the service management process.

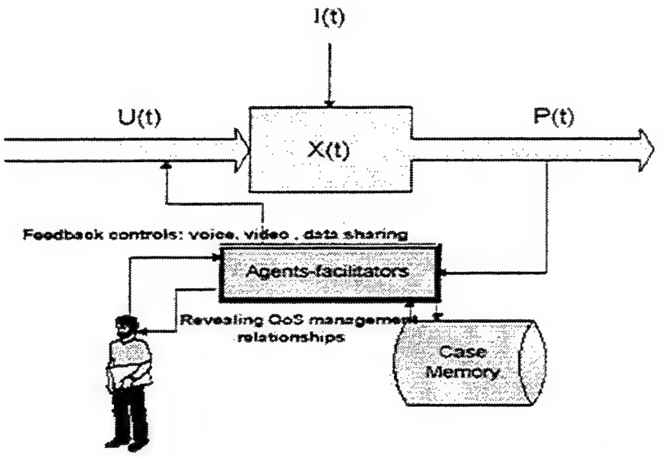

I(t)
U(t)
X(t)
P(t)
Feedback controls: voice, video , data sharing
Agents-facilitators
Revealing QoS management relationships
Case Memory

Figure 3.5. Feedback Control Model for Individual Agent Adaptation. From [Bordetsky].

The memory architecture of *agents-facilitators* is layered and divided into bridge or router agents operating with different combinations of feedback control layers. Objects such as individual collaborator profiles, QoS indices for multimedia streams and timely events, problem solving task profiles, and other collaboration objects form different layers of case frame representation. The agent-facilitators enable collaborators to communicate via desktop video conferencing and shared applications at different levels of bridges, routers, and gateways, depending on which segments of case memory are involved. [Bordetsky]

The router agent plays a major role in providing feedback controls and adaptation in service management. First of all, it provides user memory transactions by capturing the necessary information to support personal, document, and task profiles. Second, the router agent helps locate appropriate human sources of knowledge and manage desktop video conferencing calls to selected experts. Lastly, it provides for the training and capturing of QoS management knowledge in case memory. Figure 3.6 illustrates the feedback control association of service process outputs and SLM process state variables with user input controls into the case memory. [Bordetsky]

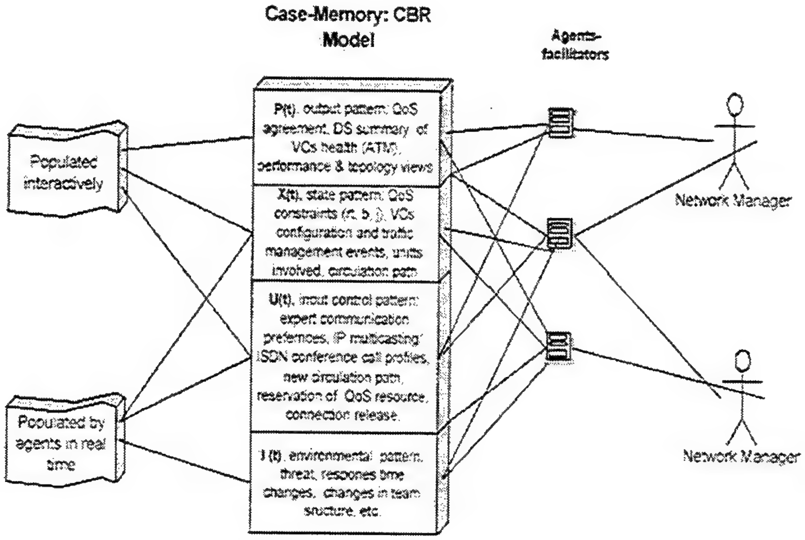

Case-Memory: CBR Model
Agents-facilitators
Populated interactively
Populated by agents in real time
P(t). output pattern: QoS agreement. DS summary of VCs health (ATM), performance & topology views
X(t). state pattern: QoS constraints (r, b, j). VCs configuration and traffic management events, units involved, circulation path
U(t). input control pattern: expert communication preferences, IP multicasting/ ISDN conference call profiles, new circulation path, reservation of QoS resource, connection release.
J (t). environmental pattern, threat, respones time changes, changes in team structure, etc.
Network Manager
Network Manager

Figure 3.6. Feedback Control Association. From [Bordetsky].

Figure 3.6 represents the knowledge retrieval model, in which each interface between layers from the bottom-up is an association based on the underlying levels. The content profiles and user response time requirements are captured in real time and populate the lower segment of the case memory stack. The agents capture the sequence of application calls (content profile) with corresponding time stamps and convert them into response time and bandwidth requirements that populate the QoS segment of a case memory frame. [Bordetsky]

In general, the QoS constraints associated with a specific SLM event are comprised of boundaries that define preferred bandwidth for voice, video, white board, and application sharing. According to such profiles, each conferencing node has associated voice, video, whiteboard, and/or application sharing delivery trees. Switching among these delivery trees helps to satisfy otherwise infeasible response time requirements. Moreover, the rules for switching delivery trees can vary based on the system, such as operational heuristics, for example. Each SLM event has a corresponding set of rules that is associated with the QoS segment of the agents' case memory. The router agent reads the QoS segment of the feedback control association from the case memory and coordinates the delivery tree switching (i.e. bandwidth allocation solutions) with the other agents-facilitators. When coordination is done, the agents transfer the coordinated solution to the network layer connection control.

3. Connection Control Adaptation: NE Layer Adaptation

On the lower network element layer, the Connection Control requirements include [Bordetsky]:

- Supervising QoS parameters
- Providing flow control, congestion control, routing, reservation and renegotiation of services
- Modifying and releasing connections
- Notifying applications to allow them to adapt

As opposed to Call Preparation Control, in which decisions are made *before* the call is made, Connection Control is done on an ongoing basis throughout the duration of the call. Feedback regarding network conditions is continuously collected and processed to allow the applications in use to adapt. Being that the most dynamic network resource is allocated channel bandwidth, this becomes the targeted area for network layer feedback controls.

H. SUMMARY

This chapter discussed the fundamental concepts of adaptive QoS management before continuing with the development of the proposed agent framework. The proposed agent framework combines the concepts of case based reasoning, service level management, policy based management, and telecommunications management network, in order to optimally solve a dynamic problem. The proposed agent framework is built on the concept of multi-layered feedback, where feedback is defined in terms of CPC, long-terms adaptation, or CC, short-term adaptation in order to account for learning and decision-making, while best transferring the service level requirements of the warfighter.

IV. PACIFIC REGION NETWORK OPERATIONS CENTER (PRNOC)

In this chapter, we analyze the operations of the Pacific Region Network Operating Center (PRNOC) in Wahiawa, Hawaii as the real-world focal point of the study. Building on the concepts in the first three chapters, we gather the empirical data necessary to determine particular user patterns and thereby derive specific application requirements for resolution by adaptive QoS management via agent technology. This chapter encompasses the basic operating principles of the PRNOC for bandwidth management, the standard network configuration, services provided for warfighters, and user loads for various C4I network applications. In Chapter V, we utilize these operating principles in conjunction with the proposed agent framework to investigate the feasibility of implementing agent technology at PRNOC.

A. PACKETSHAPER

PRNOC utilizes Packeteer's Packetshaper as its bandwidth management tool to manually enforce policy-based bandwidth allocation for network services (applications). This tool classifies and analyzes network traffic and generates a wide range of statistical charts and reports. In general, a Packetshaper is inserted in the network path and is transparent to the devices it is placed between, depending on the mode (Figure 4.1). If the *Shaping mode* is turned on, the Packetshaper influences the network traffic that passes through it. Otherwise, it only monitors in the *Monitor mode.* The Packetshaper can be characterized as a Layer 4 or Service Layer network management manager. It keys on the

TCP connections that pass through it and analyzes and manages the connections to perform its bandwidth allocation. It has direct influence on the endpoints of the TCP connection, the client and server, as opposed to a router which implements/enforces policy on directly connected interfaces only. [PRNOC CONOPS]

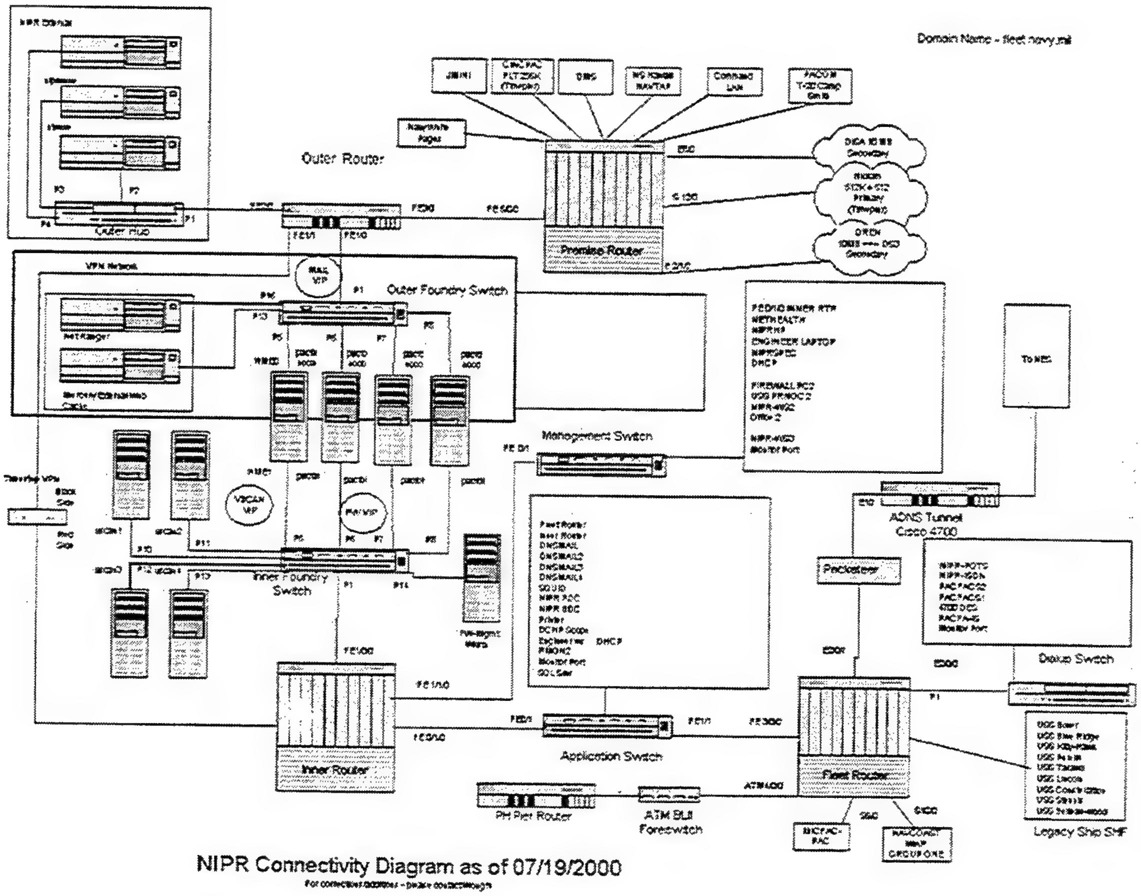


Figure 4.1. PRNOC NIPR Connectivity Diagram. From [PRNOC].

As part of network traffic classification and analysis, the Packetshaper in *Discover mode* will recognize, identify by name, and create subclasses for an extensive list of network services. It can also provide real-time throughput monitoring as well as collect data for time series and other data analysis. The real-time monitoring is good for feedback on current performance and troubleshooting. The time series data is valuable for baselining, profiling, capacity measurements, planning, and trend analysis. [PRNOC CONOPS]

1. **Packetshaper Key Features** [Performance Specification]:

➢ *Traffic Classification.* Classify traffic by application, protocol, port number, URL or wildcard, host name, LDAP host lists, DiffServ setting, IP precedence bits, IP or MAC address, direction (inbound/outbound), source, destination, MIME type, web browser, Oracle database, and Citrix published application. Detect dynamic port assignments, track transactions with migrating port assignments, and differentiate among applications using the same port.

➢ *Response-Time Management.* Track response times, divided into server and network delays. Identify the clients and servers with the slowest performance. Find out who generates or receives the most traffic of a given type. Discover the percentage of bandwidth wasted by retransmissions. Correlate dropped packets with their corresponding applications or servers. View over 30 other measured variables.

➢ *Service Level Agreements.* Set response-time commitments in milliseconds. Measure and track service-level compliance.

➢ *Partitions.* Protect or cap all the traffic in one class. Specify the size of the reserved virtual link, choose if it can grow, and optionally cap its growth. Partitions function like frame relay PVC's, but with the added important benefits that they cost less and they share their unused excess bandwidth with other traffic.

➢ *Rate Policies.* Keep greedy traffic sessions in line or protect latency-sensitive sessions. Deliver a minimum rate (perhaps zero) for each individual session of a traffic class, allow that session prioritized access to excess bandwidth, and set a limit on the total bandwidth it can use.

2. **Packetshaper Sizing Considerations** [PRNOC CONOPS]

➢ *Aggregate throughput.* The Packetshaper is placed in-line in the stream of traffic to be monitored and/or shaped. It thus must be capable of monitoring/shaping the volume of traffic passing through it. PRNOC currently uses Packetshaper 1500's with 1M inbound/outbound rates on the classified side, and 4500's with 5M inbound/outbound on the unclassified side.

- *Network connection.* The Packetshaper network connection is 10/100 Mbps Ethernet. Adapters are needed if the backbone is fiber.

- *Limited number of classes.* Packetshapers deal with entities known as classes. Hosts or subnets are classes. Networks services under the hosts are also classes (subclasses under the host). The PacketShaper builds a tree of classes and subclasses. As there are limits to the aggregate throughput that a given PacketShaper can handle, there is a limit to the number of classes to each PacketShaper.

- *Number of Top Talkers/Top Listeners.* This is limited to 12 talkers/listeners. Once the limit is reached, an older one must be dropped in order to use one for a new class.

- *Time limit for data collection.* The PacketShaper will collect data for up to 60 days at which point data will be dropped off.

3. Accessing the Packetshaper

There are three methods to access the Packetshaper [PRNOC CONOPS]:

- *Web mode.* The web mode is the most popular mode. Access is via a web browser. In the web mode, login is either *to touch mode* or *to look mode.* Once accessed, the two most used screens are the Manage and the Monitor screens. The Manage mode for configurations and some reports, and the Monitor mode to view current activity. The Monitor mode has toggles for immediate update or periodic update.

- *CLI, or Command Line Interface mode.* This is reached by telnetting to the Packetshaper. A UNIX-like interface is provided. Functions available by the browser interface are also accessible by the CLI. This is useful for troubleshooting and writing command scripts to effect changes automatically without accessing the browser. Likewise to retrieve data via script (This is the native interface via console).

- *API interface mode.* This is a programmatic interface through the web browser port. This interface has not been exploited by the NOCs yet.

B. NOC ENVIRONMENT – BACKGROUND

The Fleet NOCs are the fleet portals to the Defense Information Systems Network (DISN), both NIPRNET (unclassified) and SIPRNET (secret). The NOCs provide firewall, mail store and forward, web caching and other network services There are several distinct groups of ship communications back-ended to the NOC including:

- Automated Digital Networking System (ADNS) RF (wireless, Radio Frequencies)
- Legacy RF (usually limited to SHF)
- Pier/Backhaul
- Dial-In

Packetshaper bandwidth management is primarily directed at ADNS RF. Besides the lower bandwidths available, the PRNOC has observed that unmanaged circuits will have their circuit capacities fully congested and exceeded. When the circuit is unmanaged, all service connections, whether critical or not, compete evenly for the available bandwidth. [PRNOC CONOPS]

1. Automated Digital Network System (ADNS)

ADNS is the backbone to the Joint Maritime Communications System. It uses off-the-shelf protocols, processors, and routers to create a robust and flexible networking environment. Internet Protocol (IP), Asynchronous Transfer Mode (ATM) and other products are being adopted or adapted from the commercial telecommunications world. Interfaces to all radio frequency media from High Frequency (HF) to Extremely High Frequency (EHF) provide the total throughput and access needed. At the same time, networking techniques attempt to make efficient use of all available channels.

ADNS is a unique system in which the basic backbone for all message classifications is GENSER (classified). To obtain UNCLAS or TACINTEL, encryption devices are used to encrypt the information to get to the baseline GENSER and decrypt at the other end. The bandwidth is shared among the three classifications (as opposed to separately allocated bandwidths). Certain minimum bandwidths can be guaranteed for each level (the Bandwidth Reservation System (BRS) allocation). However, bandwidth management cannot be solely considered from one level, but in totality of all three levels. Thus, if in the future, there is an increase in regular GENSER traffic, adjustments to bandwidth allocation may need to be made to both the GENSER and UNCLASS sides. The BRS allocation is set in the ADNS routers. If bandwidth requirements max out and exceed circuit capacity, then the router discards packets. Packetshaping is an effort to avoid this. [PRNOC CONOPS]

C. BASIC BANDWIDTH MANAGEMENT POLICIES

1. For ADNS Circuits, Keep (UNCLAS) MSS Size at 1200 or Below

ADNS uses both NES encryption and IP tunneling for transmitting UNCLASS traffic through the GENSER backbone. In the NOC architecture, packet fragmentation occurs if the normal MTU size of 1500 is used. Packet fragmentation has been found to cause severe throughput problems for NESs and is therefore to be avoided. This has generally meant setting MTU sizes smaller at the NOC servers that interact with fleet unit servers, and setting router MTU sizes to be smaller as well. The Packetshaper offers a more elegant solution, whereby the MSS (TCP payload) size can be set for TCP sessions where the packets traverse the Packetshaper. [PRNOC CONOPS]

2. Do Not Exceed Circuit Capacity

This means that the amount of traffic is not to exceed a certain bandwidth limit. This can be readily set on the Packetshaper. As described earlier, ADNS bandwidth is not split into separate TACINTEL, GENSER, and UNCLAS pipes. Thus, the limit is at best an approximate. Also, the bandwidth may change due to the addition or reduction of resources, and thus the limits need to be adjusted accordingly.

3. Provide Enough Bandwidth for Mail

Set aside enough bandwidth so that mail does not significantly backlog both going to and from the ship.

4. Provide a Nominal Bandwidth for DNS

5. Allow Remainder of Available Bandwidth for Webbing

6. Restrict Lotus Domino Bandwidth to Subs (GENSER)

When subs establish contact, the Domino Replication will tend to take an inordinate portion of bandwidth, causing slower mail delivery.

7. Dampening: FTP, RealAudio, RealVideo

Limit bandwidth so that it does not consume inordinate amount of current bandwidth.

To accomplish the policies set forth above, Packetshapers are deployed at the NOC in the following strategic locations:

UNCLAS	**Model**	**In/Out Rate**	**Location**
Packetshaper	4500	5M/ 5M	Between Fleet and Tunnel Router
Packetshaper2	4500	5M/ 5M	Between App Switch and Fleet Router
GENSER			
Packetshaper	1500	1M/ 1M	Between Fleet and ADNS CISCO Router
Packetshaper2	1500	1M/ 1M	Between App Switch and Fleet Router

Table 4.1. Packetshaper Locations. From [PRNOC CONOPS].

D. MONITORING/SHAPING

1. Monitoring

Monitoring is a valuable tool even without the shaping. Some of the more common uses are [PRNOC CONOPS]:

- *Baselining throughput and activity to a ship.*
- *Confirming/Verifying specific network services to/from a ship.*
- *Identifying activity from specific workstations on a ship.* This is useful in troubleshooting when problems with specific services are encountered. Sometimes it helps the ship isolate their problem to point out where network service requests are coming from and going to.
- *Time Series Analysis.*
- *Reports/comparison of Classes.*
- *User feedback loop.*

The user (ship) is also allowed to access the Packetshaper Monitoring features so that it can monitor its own activities. It can then decide whether to curtail an activity such as webbing, or review the types of offship activity occurring.

2. Shaping

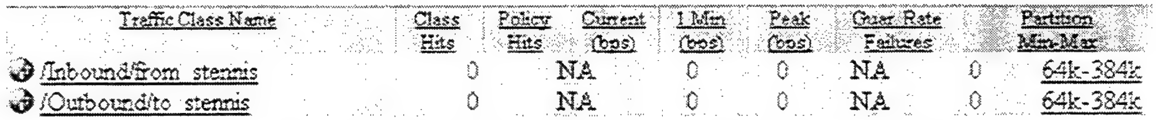
Monitoring only allows one to study the traffic. Shaping is needed to implement the policies.

F. USING PARTITIONS TO IMPLEMENT THE POLICIES

PRNOC uses the Packetshaper implementation of partitions to implement the flow management policies. With Packetshaper, PRNOC has the ability to create classes to represent ships and their network services, both inbound and outbound. For each of these classes, they can then create a virtual pipe with reserved bandwidth flow for that class. This virtual pipe is a partition. The following are ways that PRNOC uses these partitions to implement its basic bandwidth policies [PRNOC CONOPS]:

1. Do Not Exceed Circuit Capacity

PRNOC considers inbound and outbound traffic separately. They are primarily interested in the UNCLAS circuits, although individual circumstances may require focus on the GENSER circuits. In the following example, the USS STENNIS has 384K bandwidth (UNCLAS) each way.



Traffic Class Name	Class Hits	Policy Hits	Current (bps)	1 Min (bps)	Peak (bps)	Guar Rate Failures		Partition Min-Max
/Inbound/from stennis	0		NA	0	0	NA	0	64k-384k
/Outbound/to stennis	0		NA	0	0	NA	0	64k-384k

Figure 4.2. Inbound/Outbound using Packeteer. From [PRNOC CONOPS].

The Packetshaper will throttle traffic back for STENNIS Inbound as the limit is approached. Since PRNOC set the limit to 384K, if the UNCLASS rate runs at about the max rate, and the GENSER is also very active, the limit could be exceeded. This is precisely where intelligent agents from this research can help in providing optimal bandwidth allocation. Generally, PRNOC considers the maximum circuit rate, and the steady state rate that the ship seems to use. In turn, they provide a cushion of 32 to 64k over the steady state rate but not to exceed the max rate. Currently, they use a rule of thumb of about 32k for the steady state rate for GENSER and measure it with a Packetshaper on the GENSER side. [PRNOC CONOPS]

2. Smaller Bandwidths

There is greater flexibility with larger bandwidths such as the 384k illustrated above. With the smaller bandwidths such as 256K or 128K, there is less flexibility, and it is more likely to have to back down from the actual maximum of the circuit.

At this point, it is important to note that there is a difference between PRNOC and Indian Ocean Region NOC (IORNOC) in the quantity of ships that can potentially be shaped. IORNOC is more able to shape all ships chopped to it since there is a limited number of ships in the region at a time. On the other hand, PRNOC cannot shape for all ships and must select the ships it shapes for. Along with this, the aggregate bandwidth committed to partitions must be considered. Thus PRNOC will set the initial partition size lower, and let it grow as necessary. Table 4.2 is a spreadsheet of the partition settings for three large decks in PAC. Besides the Outbound and Inbound, partitions for services SMTP, HTTP, and DNS are shown. [PRNOC CONOPS]

Current Packetshaping for Lincoln, Stennis and Vinson at PRNOC						
				Partition	(All Burstable)	
Lincoln				Min	Max	
Inbound (from Lincoln)				32k	128k	
	HTTP			0	64K	
	SMTP			64k	128k	
	DNS			1000	32k	
Outbound				64k	196k	
	HTTP			0	152k	
	SMTP			64k	96k	
	DNS			1000	48k	
Stennis				Min	Max	
Inbound (from Stennis)				64k	384k	
	HTTP			0	256k	
	SMTP			64k	96k	
	DNS			1000	32k	
Outbound				64k	384k	
	HTTP			0	256k	
	SMTP			64k	96k	
	DNS			1000	32k	
Vinson				Min	Max	
Inbound (from Vinson)				64k	384k	
	HTTP			0	172k	
	SMTP			64k	128k	
	DNS			5000	20k	
Outbound				96k	274k	
	HTTP			0	172k	
	SMTP			96k	172k	
	DNS			5000	20k	

Table 4.2. Packetshaping for Carriers. After [PRNOC CONOPS].

3. Small Decks

All the preceding discussion centered on large decks (i.e. carriers and ARGs) because PRNOC has limited resources. Consequently, PRNOC has primarily been shaping the large decks. Alternatively, IORNOC has also shaped the small decks since it has fewer ships in its operating area at a time.

4. Inbound/Outbound Partition Sizes

There is also a partition for the entire Inbound/Outbound Class. Its maximum size is somewhat smaller than the rate size for the Packetshaper. The Packetshaper will warn the user if he tries to set it too high.

5. Providing Enough Bandwidth for Mail (SMTP)

Unmanaged, mail tends to get bogged down when there is high web activity. By providing a large enough partition for mail, a good mail delivery rate can be maintained. Inbound and outbound are handled separately. Outgoing (to ship) generally is of higher volume than Incoming (from ship). PRNOC uses the following approach: (1) Set the initial partition size and limit, and monitor the traffic during a period of normal use; (2) If the rates seem to bump up around the partition limit, increase the limit; and (3) If the rates are somewhat lower than the limit, lower the limit. [PRNOC CONOPS]

As another gauge for Outgoing (to ship), PRNOC monitors how quickly the mail queues process, whether they tend to back up, move slowly out, or move quickly out. For Incoming (from ship), besides watching the rate, PRNOC rates feedback from the ship as a useful indicator as to whether their mail off ship is being sent at a decent rate. Again, at the lower throughput rates, there is less leeway on the partitions settings. At the lower bandwidths, mail generally takes about 40 per cent, and web takes 60 percent of the traffic bandwidth. At the higher bandwidths, more webbing can be performed, as mail does not increase proportionately.

6. Burstable

Setting the partition as burstable means that that bandwidth can be taken from another partition if the other partition is not using it. Thus web can use the SMTP unused bandwidth if it needs to. When SMTP needs more bandwidth it can regain it.

7. Provide Nominal Bandwidth for DNS

DNS is an important service mainly in that without its proper operation, webbing (HTTP) does not work.

8. Allow Remainder of Bandwidth for Webbing

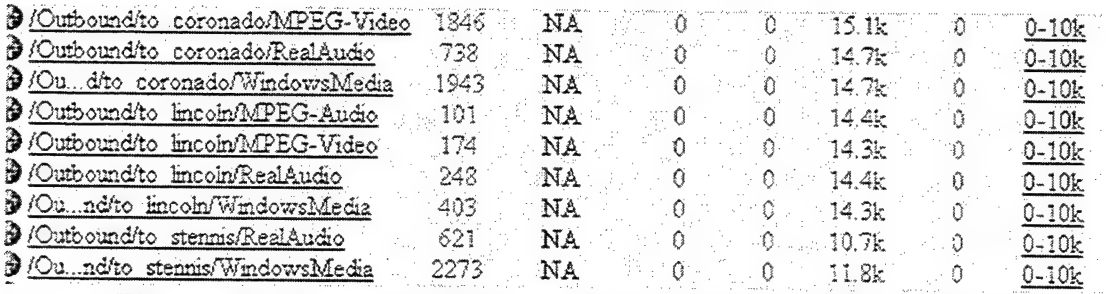
Once SMTP and DNS have been accommodated, the remainder can be used for webbing. Essentially, it will be webbing that will be throttled back if the pipe gets congested.

9. Restrict Lotus Domino Bandwidth to Subs (GENSER)

When subs establish contact, the Domino replication will tend to take an inordinate portion of bandwidth, causing slower mail delivery.

10. Dampening: FTP, RealAudio, RealVideo

Limit bandwidth so that it does not consume an inordinate amount of current bandwidth. Figure 4.3 illustrates services that are sharply dampened because they have the potential to heavily consume the bandwidth if unchecked. Alternatively, they can be blocked entirely. Most are streaming media. The partition limit is 10k for each.



/Outbound/to coronado/MPEG-Video	1846	NA	0	0	15.1k	0	0-10k
/Outbound/to coronado/RealAudio	738	NA	0	0	14.7k	0	0-10k
/Ou...d/to coronado/WindowsMedia	1943	NA	0	0	14.7k	0	0-10k
/Outbound/to lincoln/MPEG-Audio	101	NA	0	0	14.4k	0	0-10k
/Outbound/to lincoln/MPEG-Video	174	NA	0	0	14.3k	0	0-10k
/Outbound/to lincoln/RealAudio	248	NA	0	0	14.4k	0	0-10k
/Ou...nd/to lincoln/WindowsMedia	403	NA	0	0	14.3k	0	0-10k
/Outbound/to stennis/RealAudio	621	NA	0	0	10.7k	0	0-10k
/Ou...nd/to stennis/WindowsMedia	2273	NA	0	0	11.8k	0	0-10k

Figure 4.3. Streaming Media Applications. From [PRNOC CONOPS].

F. OTHER CONSIDERATIONS

1. Ships with Multiple Servers

Since the class extends across all the networks on the ship, all servers of a particular server (such as SMTP) will share the same bandwidth in the partition. This would be the case for an ARG with Navy and Marine units on board, or a carrier with ships crew and staff on board. If the rate of mail is significantly larger due to this, then there needs to be an increase in the partition limit for SMTP or whatever service needs it.

2. Ships with Multiple Links

Some ships may have multiple links, such as dual HSD. If this is a somewhat permanent situation, PRNOC recommends considering the bandwidth to be sum of the two links.

3. Changing Bandwidth

It is important that the NOC be aware of any bandwidth changes for the ship. If the bandwidth is increased, it means that the circuit is not optimized for that bandwidth. Worse, if the bandwidth is decreased, it may mean exceeding the circuit capacity.

V. AN AGENT SOLUTION FOR PRNOC

In this chapter, we explore an agent-based solution to the QoS dilemma at PRNOC. First, we review the various levels of QoS that may benefit from agent technology. Then, we develop a potential agent solution for one specific aspect of the QoS problem; that is, bandwidth allocation within the UNCLAS network. This solution is based on an agent-technology/Packetshaper partnership. In this partnership, the agent structure is overlapped on the network management infrastructure (HP Openview) and works in a two-way feedback loop with the Packetshaper. With agent coordination, the two components work hand-in-hand to intelligently manage the bandwidth allocation problem.

A. PRNOC QOS DILEMMA

Based on the information from the previous chapter, there are various areas where PRNOC can benefit from adaptive intelligent decision-making to tackle its dynamic bandwidth allocation problem. While PRNOC has undertaken significant measures in tackling the bandwidth allocation problem, at the same time, they recognize certain shortfalls that need to be addressed. As stated in the previous chapter, three message classifications (UNCLAS, GENSER, TACINTEL) share one bandwidth pipe, in which bandwidth management is not solely considered from one level, but in totality of all three levels. As a further note, each ship has its own bandwidth allotment and does not compete with other ships. This bandwidth allotment can change due to changing resources or capabilities.

Bearing these factors in mind, there are various levels of QoS management that need to be addressed. The first level is managing the QoS requirements for each respective classification. For example, in the UNCLAS classification, there are numerous applications including HTTP, SMTP, DNS and many others that compete for the bandwidth allocated for that portion of the overall pipe, i.e. 33%. PRNOC uses the Packetshaper functionality of partitions to divide that portion of the pipe in accordance with historical usage patterns as discussed in Chapter IV. Certain applications like HTTP, SMTP, and DNS get specified allocations, while the rest basically fight for what is left, (with certain streaming media applications dampened for control purposes). Unfortunately, the dampening of certain streaming media applications limits their usage even if there is capacity available elsewhere. Moreover, changing partitions requires manual monitoring and inputs. This chapter proposes a solution whereby agents provide intelligent adaptive capability to maximize allocation.

The second level of the QoS dilemma involves the ship's entire bandwidth allocation, i.e. all three levels. As the system works now, when the UNCLAS portion exceeds its 33% allotment, it taps into the GENSER allotment, provided there is room. The problem arises when the demand for GENSER increases such that there is no additional room for UNCLAS usage. When this happens, all UNCLAS "infringements" are immediately cut-off, which obviously creates a frustrating situation for any ongoing UNCLAS applications. Instead, the desire would be for a more gradual back-off solution.

With agent technology, agents can conceivably coordinate all three classifications to maximize the bandwidth usage. The set-up would entail separate Packetshapers for each classification, in which each Packetshaper would work with the agents in a feedback relationship. Packetshapers are already in place for the UNCLAS and GENSER networks, but not for TACINTEL. With an agent structure to coordinate the usage of each portion of the pipe and an overarching agent structure to coordinate the whole, bandwidth usage can be maximized for sharing. More specific working arrangements for a specific portion (UNCLAS) are discussed later in the chapter.

A third level of the QoS dilemma is probably the most challenging, but also the most useful if ever developed. This level would entail adaptive on-demand prioritization of specific applications for a specific classification. For example, suppose a ship required maximum UNCLAS HTTP above all, or maybe unlimited bandwidth for GENSER applications like Lotus Domino web replication. This would require a high degree of coordination among all classifications. The prioritization of such an application would have to be applied and recognized across the network and across classification boundary lines, which is not an easy proposition because of the network configuration. As discussed in Chapter IV, the basic backbone is GENSER. To obtain UNCLAS or TACINTEL, encryption devices are used to encrypt the information to get to the baseline GENSER and decrypt at the other end. This makes it all the more difficult to make a high priority request, wherein all network elements of all classification levels are made aware. In short, an agent solution would have to be very complex to ensure the proper coordination of such requests.

B. DATA

In order to keep this thesis unclassified, the research is based on bandwidth utilization data from the UNCLAS ADNS Packetshaper at PRNOC (APPENDIX). Other data types were gathered from the Packetshaper including round-trip delay, packets per second, and total bytes, but not analyzed for this thesis.

This research focuses on the first layer of the bandwidth utilization QoS dilemma discussed above. However, the solutions developed for UNCLAS are just as applicable to the other classification categories.

1. Data Limitations

Packetshaper is a relatively new tool that has been used at PRNOC for less than one year at the time of this research. With further study, data gathered from this Packetshaper could greatly help PRNOC to develop better bandwidth management policies. As it stands, these policies are based on intelligent guesses, experience, and trial and error.

Currently, the data gathered at PRNOC has several limitations. First, it only applies to ships operating in the Pacific region. This is limiting if one desires to develop bandwidth usage patterns for a ship or ship class throughout its entire operational schedule, (i.e. work-ups, deployment). In order to conduct such a study, additional information would be needed from other NOCs since deployments are generally not confined to the Pacific region. Second, the data is only stored for two months due to lack of storage space and funding. As this kind of bandwidth study earns more recognition, more funding could occur in the future. Third, due to the class limitations of PRNOC, not all ship classes can be adequately monitored with Packetshaper. Moreover, for each ship class, not all applications can be accounted for. Fourth, application measurement is generally reserved to at-sea operations. In port, there are varying levels of pier connectivity. Fifth, it is hard to effectively generate or validate application usage tendencies based on the limited data. Each ship had different capabilities and operated at different phases of their operational schedule. Only with more data could this ultimately be accomplished.

2. Data Trends

The data gathered was based on three operational amphibious ships: USS BELLEAU WOOD (LHD-3), USS TARAWA (LHA-1), and USS BOXER (LHD-4). At the time of data gathering, these ships had the widest amount of data for more application types for one particular ship class. For each ship, data was gathered for Outbound and Inbound, including total bandwidth utilization, HTTP, SMTP, DNS, Windows Media, Real Audio, MPEG Audio, and MPEG Video. Data was gathered for the last four applications with the view that they could become more useful in the future if given the bandwidth.

In general, the data shows UNCLAS application traffic that remains within its allotment for the most part, but pushes the limits in certain instances on BOXER and BELLEAU WOOD. Moreover, for BOXER and BELLEAU WOOD, where no partition policies are in place, there are several large bandwidth utilization spikes for the streaming media applications. With respect to the bandwidth utilization as a whole, the data affirms the tendency for more data to go *to* a ship than come *from* a ship. What can be gathered from this data lends credence to the bandwidth policies and experience at PRNOC. While the data is not enough to fully categorize application patterns for a certain class of ship under all levels of operation, the data does demonstrate a few important generalizations. First, the data clearly shows the need for application bandwidth management. Without it, there is no prioritization and a lack of control among the various applications. Second, bandwidth utilization for certain streaming media applications spike significantly on certain occasions, demonstrating the need for specific attention while at the same time showing that more utilization would take place if there were more bandwidth. In this area, intelligent agents could conceivably calculate the best tradeoffs among high bandwidth consumption, priority, and overall bandwidth availability. Third, the data can be highly useful in determining user patterns if combined with operational data. For example, if bandwidth utilization increases significantly whenever a ship is involved in a certain operation, than this kind of information could become very valuable in establishing priorities for future usage. In turn, this could be programmed into the agents' knowledge base.

C. AGENT SOLUTION

1. Methodology

The methodology for understanding the problem and subsequently deriving an agent solution is discussed in Chapter III. In essence, the basic goal is to derive an agent framework that intelligently makes decisions in the dynamic allotment of bandwidth. The first step is to gather requirements, from which application priorities can be derived. PRNOC has already documented some of its efforts to understand the bandwidth allocation problem. With further study, an agent solution can be expanded to other than peacetime operations. Based on the data gathered from this study, intelligent agents can clearly aid in this problem.

The second step is to utilize service level management (SLM) techniques to effectively understand and translate requirements between the network service provider and warfighter. Based on the current situation, it is clear that there is a mutual understanding between the two camps as far as priority, i.e., e-mail, HTTP, etc. As such, the specific goals for the agents are clear. Third, with policy-based management (PBM), the policies can be derived and thereby translated into the agent knowledge base. Again, PRNOC has already derived general policies based on experience, but not so for the classified networks and higher operational tempo operations. Agent technology can increase the timeliness and efficiency in implementing them. Fourth, developing an implementation plan that accounts for interfaces and interoperability. With a partnership between Packetshaper and HP OpenView already in place, adding the agents to serve as the liaison between the two is highly feasible. Fifth, implement the change and monitor progress. The agents much make decisions quick enough to be useful without taking up too much of the network's memory resources.

2. Learning

Learning is one of the more important capabilities afforded by the proposed agent solution. The case based reasoning library is ideal in the development of agent actions to dynamic bandwidth requests. As historical data is collected, the case library is continually updated, further enhancing and speeding up the agents' ability to react. With learning, partition sizes can be updated by the agents instead of manually. Moreover, the agents can develop patterns to better predict future operations.

3. Deriving Agent Committees

The data acquired at PRNOC can be used to develop operational heuristics for agent committees. These committees would replace or enhance the manual policy implementation currently taking place. Different agent committees can have different voting priorities based on the situation or environment. For example, based on most of the experience at PRNOC, e-mail is the number one priority, followed by web and domain name service (DNS). As such, e-mail gets the largest allotment of bandwidth, the other two applications receive guaranteed allotments, and the rest have no priority. While this policy is satisfactory in meeting non-deployment operations in the Pacific region, it does not account for a changing environment.

To satisfy this problem, agent committees could be changed out to meet ship priorities. For example, in wartime, the priorities may switch to streaming media in addition to e-mail. This "wartime" committee would have a different voting structure to better prioritize bandwidth to meet the situation. In addition to peace or war, other committees can be derived to meet specific operations with specific requirements.

In the future, streaming media applications should be more significant in the future. However, the ability to change out agent committees is probably more significant on the GENSER side, wherein more tactical communications take place. Furthermore, on the GENSER side, there are many more bandwidth intensive collaborative applications that could take advantage of this capability in order to better utilize the bandwidth and accommodate more sharing.

D. AGENT IMPLEMENTATION

The agent setup will be overlapped onto the HP OpenView network management functionality. One benefit of Packetshaper is that it is designed to interface with HP OpenView. Consequently, agents have an interface with Packetshaper via HP OpenView. Agents can be placed on the client machines or the central monitoring station in the HP OpenView network. In this manner, they can either represent functional capability or local machines. In other words, agents can each control applications as HTTP, SMTP, etc. from all client machines; or, agents can represent numerous application types from each particular client machine. In this thesis, we choose the former distribution of agents. In this framework, the agents will all be physically located on the HP OpenView managing computer. In the final analysis, either way can feasibly work, but bandwidth allotment would probably be better represented by agents aligned to represent applications, as opposed to agents having a particular affiliation with certain client machines.

Based on the information received from the Packetshaper, agents can affect traffic levels at the source before they reach the Packetshaper. In other words, the agents bridge the service level requirements with the network management layer requirements via HP OpenView. At the same time, the agents correspondingly direct the Packetshaper to shape or throttle traffic. In sum, the agent partnership is based on the concept of feedback, whereby the agents provide the go-between via HP OpenView and Packetshaper.

With respect to the agents themselves, they vote in accordance with their committee voting priorities as discussed above and work in accordance with the artificial neural network discussed in Chapter II. The experience base at PRNOC will provide the initial foundation for the case library. ZEUS agents or Proteus agents are probably the best toolkits for this application to start.

VI. CONCLUSIONS AND RECOMMENDATIONS

As postulated in such documents as Joint Vision 2010/2020 (JV2010/2020), the Concept of Future Joint Operations (CFJO), and the Revolution in Military Affairs (RMA), advances in technology and information superiority will revolutionize the way military forces operate in the 21st Century. New and improved technologies will expand the battlespace and compress the time commanders have to react to rapidly developing situations. To adapt, Joint Force Commanders (JFC) must embrace technology and establish command and control processes and procedures that maximize the technological advantages of the joint force to achieve full spectrum dominance.

> [In the 21st Century] The unqualified importance of information will not change. What will differ is increasing access to information and improvements in the speed and accuracy of prioritizing and transferring data brought about by advances in information technologies. While the friction and fog of war can never be eliminated, new technologies promise to mitigate their impact. [Mayer & Stover]

In essence, the overarching purpose of this research has been to investigate a technology that can potentially help the JFC make better decisions in the new battlespace of the 21st Century. The targeted area of this research is Quality of Service (QoS), which is a critical factor that permeates throughout the Global Information Grid. While QoS is behind the scenes in nature, its importance in information transfer is not.

A. SUMMARY

In this thesis, we investigated the feasibility of using agent technology for the adaptive QoS management of C4ISR networks. To that end, we developed a multi-agent system framework based on the attributes of *collaboration* and *adaptability*. We identified these characteristics as critically necessary to meet the dynamic and heterogeneous requirements that are typical of joint C4ISR networks. Furthermore, we based the agent framework on the concepts of Service Level Management (SLM), Policy Based Management (PBM), and Telecommunications Management Network (TMN) in order to best develop a system that optimizes customer (warfighter)/service provider (NOC) communication, facilitates complex policy implementation, and follows a prevalent framework for telecommunications networks.

With respect to the agent decision-making process, we proposed the case based reasoning (CBR) approach as an effective way to facilitate quicker, more efficient decision-making. With this approach, knowledge is continually updated and built upon through feedback and the case based library. The agents are situated in a committee structure overlaid onto an artificial neural network (ANN) structure. In this manner, simpler solutions are developed on the first layer, while complex decisions are only made in the second layer of the ANN as necessary, saving time and computing power.

Finally, we applied our proposed methodology to a C4ISR application at the Pacific Region Network Operations Center (PRNOC) in Wahiawa, Hawaii, which in turn, exposed a variety of bandwidth allocation issues that could benefit from agent technology. The PRNOC QoS dilemma proved to be a classic case for agent implementation.

B. CONCLUSION

This work is but an introduction into the endless possibilities of agent technology for adaptive quality of service management. However, the findings of this study clearly demonstrate the feasibility and advantages of developing agent technology for this purpose. Agent technology is already being explored in various forms for the adaptive QoS management of multi-media services in the commercial sector. In light of this, it is clear that the success achieved in the commercial sector can be applied to military joint C4ISR networks. As the agent paradigm continues to spread, this will have an ever-growing positive impact on this particular area of research.

C. RECOMMENDATIONS FOR FURTHER STUDY

Unfortunately, many other relevant areas could not be covered in greater detail due to time restraints. The following are recommendations for further expansion based on this thesis:

1. Continue Development of an Agent Solution at PRNOC

This thesis merely lays down the theoretical groundwork for one aspect of the bandwidth allocation dilemma at PRNOC. The next step would be to physically implement and test the agent system, which would probably be safer and less intrusive at an agent testbed before going directly to PRNOC. Such issues as which agent toolkit to use, interoperability issues, and timing issues can only be resolved in the physical implementation phase.

In addition, as discussed in Chapter V, there are also two other layers of the QoS dilemma at PRNOC that can be addressed via agent technology. While these problems technology can be used to coordinate the bandwidth usage for the entire pipe as a whole, i.e. UNCLAS, GENSER, and TACINTEL. In the third layer, agent technology could prove invaluable in solving on-demand QoS requests for specific applications within a certain classification. This represents the greatest challenge in that it requires the most coordination and communication across the all three classifications operating the GENSER backbone. While these problems are more complex, developing a solution for these two other layers should prove to be more helpful and useful to PRNOC.

2. Continue Bandwidth Usage Study at PRNOC

This thesis highlighted the usage of Packeteer's Packetshaper as an invaluable bandwidth allocation tool at PRNOC. In addition to its remarkable shaping capabilities, Packetshaper's data gathering capabilities can be highly useful in determining bandwidth usage patterns, which in turn promotes better allocation at the NOCs. More importantly with respect to agent technology, these patterns can speed the development and accuracy of agent committees. With these committees, agent technology can most accurately speed the decision-making process to a broader range of operations.

3. Modeling and Simulation

Modeling and simulation would be highly beneficial in testing the agent decision-making process. Extend by Imagine that, Inc. and OPNET, by OPNET Technologies, Inc. are two tools that can be especially useful. With modeling and simulation, it is easier to test various agent frameworks and identify problems. Modeling and simulation is a necessary step in validating the ideas of any proposed agent framework and is the next logical step for this thesis.

4. Develop an Agent Testbed

Currently, classroom work is in progress to develop an agent testbed based on some of the principles of this thesis. The agent testbed is an invaluable tool used to test various agent toolkits and schemes. As shown in Chapter II, there are numerous ways to situate agents. Actually testing each one in a testbed environment is a good way to compare the different methods in terms of logic, speed, accuracy, and effectiveness.

In the future, there is also a draft proposal to form a Center for Research in Global Information Grid Operations at the Naval Postgraduate School for advanced studies in GIG operations including agent technology and adaptive QoS management [Bordetsky GRID]. Under this proposal, the Agent Grid Testbed for GIG Adaptive Management will be an important resource in the physical testing and implementation of agents [Bordetsky GRID]. At the same time, the work will be used to support the Naval Postgraduate School's continuing research partnership with the Joint Experimentation Directorate, U. S. Joint Forces Command (J-9, JFCOM).

5. Explore Wireless Feasibility Issues

The wireless phenomena opens up a wide range of complex and important issues that can occupy a thesis topic in itself. With an ever-increasing reliance on wireless technology in the military, wireless issues must be resolved in order to successfully use agents in the adaptive management of C4ISR networks.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX. BANDWIDTH UTILIZATION DATA

This Appendix contains bandwidth utilization data obtained at the Pacific Region Network Operations Center (PRNOC) in Wahiawa, Hawaii from 26 – 29 March 2001. The graphs show bandwidth utilization data for three ships of the same ship classification (LHA/LHD) that were being tracked by PRNOC. They are USS TARAWA (LHA-1), USS BELLEAU WOOD (LHA-3), and USS BOXER (LHD-4).

For each ship, the following traffic types are represented: Inbound from ship to PRNOC, Outbound from PRNOC to ship, Outbound HTTP, Outbound SMTP, Outbound DNS, Outbound RealAudio, Outbound MPEG Audio, Outbound MPEG Video, and Outbound Windows Media. Note that inbound traffic does not push the limits of bandwidth usage. Since Inbound is not the focus of this thesis, only the *total* inbound traffic is represented. Conversely, since the outbound traffic is more bandwidth intensive and likely to push the limits, additional traffic types are represented to show the makeup and tendencies in the traffic.

The bandwidth limits for each ship are as follows: USS TARAWA: 128 kbps, USS BELLEAU WOOD: 384 kbps, USS BOXER: 384 kbps. Also note that traffic monitoring for the various ships may have started at different times, depending on PRNOC. In addition, TARAWA was inport from deployment from 14 February – 20 March and did not register any traffic. For this period, TARAWA was the only ship that had partitions implemented by PRNOC. Table A.1 shows the partitions in place at the time of data gathering.

Bandwidth Policies: **Outbound to USS Tarawa**	**Partition Size:**
Total	0-128k
HTTP	0-64k
MPEG Audio	0-20k
MPEG Video	0-20k
SMTP	32-64k
DNS	5000-10000
Windows Media	0-20k

Table A.1. Partition Sizes for USS Tarawa.

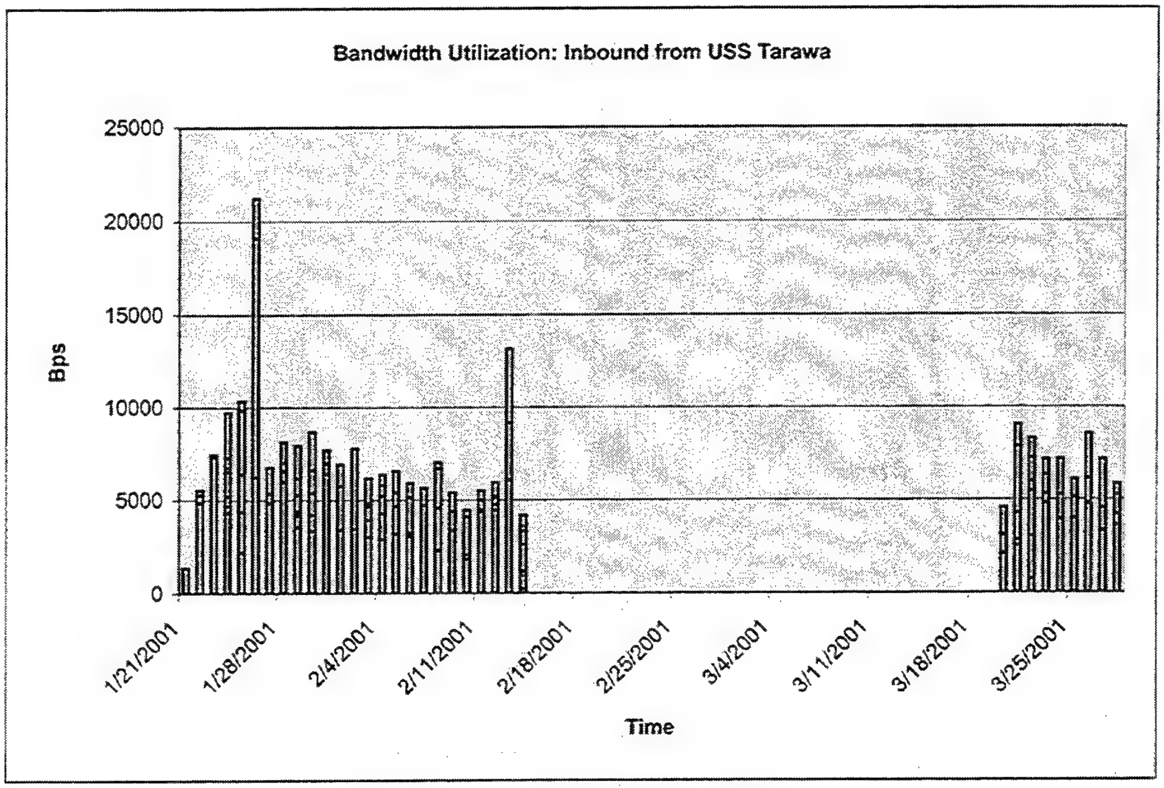


Figure A.1. Inbound from Tarawa.

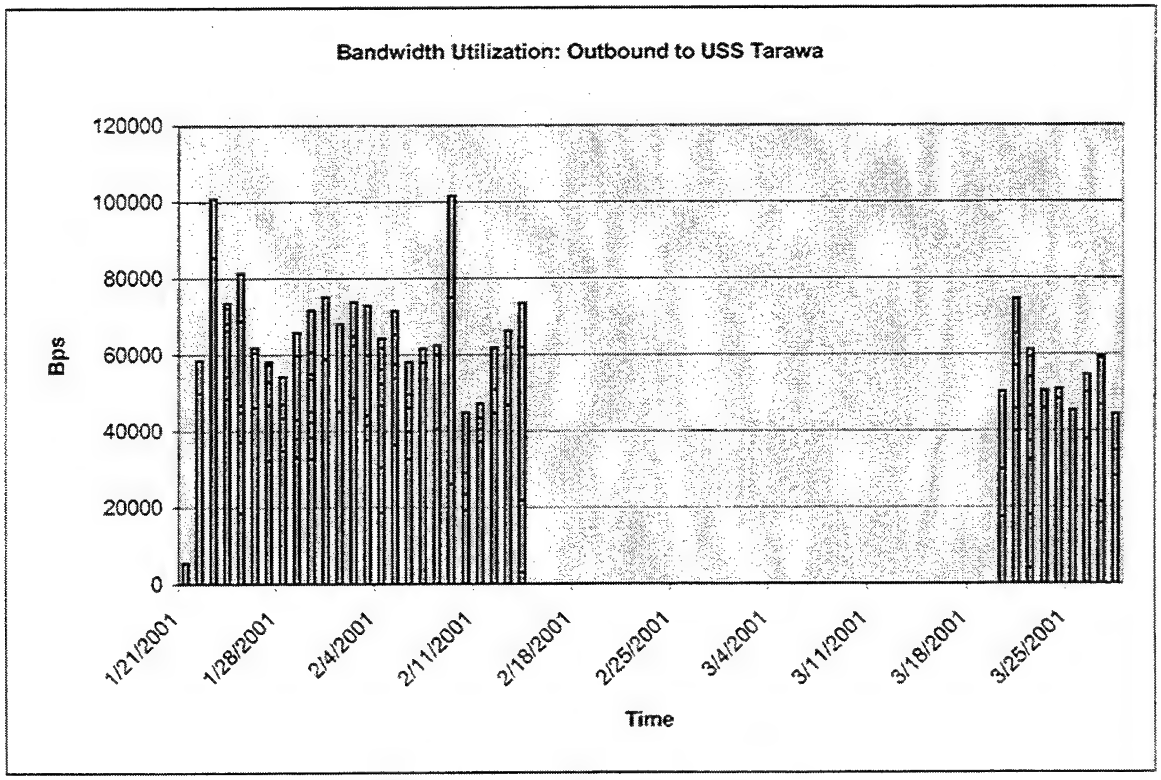


Figure A.2. Outbound to Tarawa.

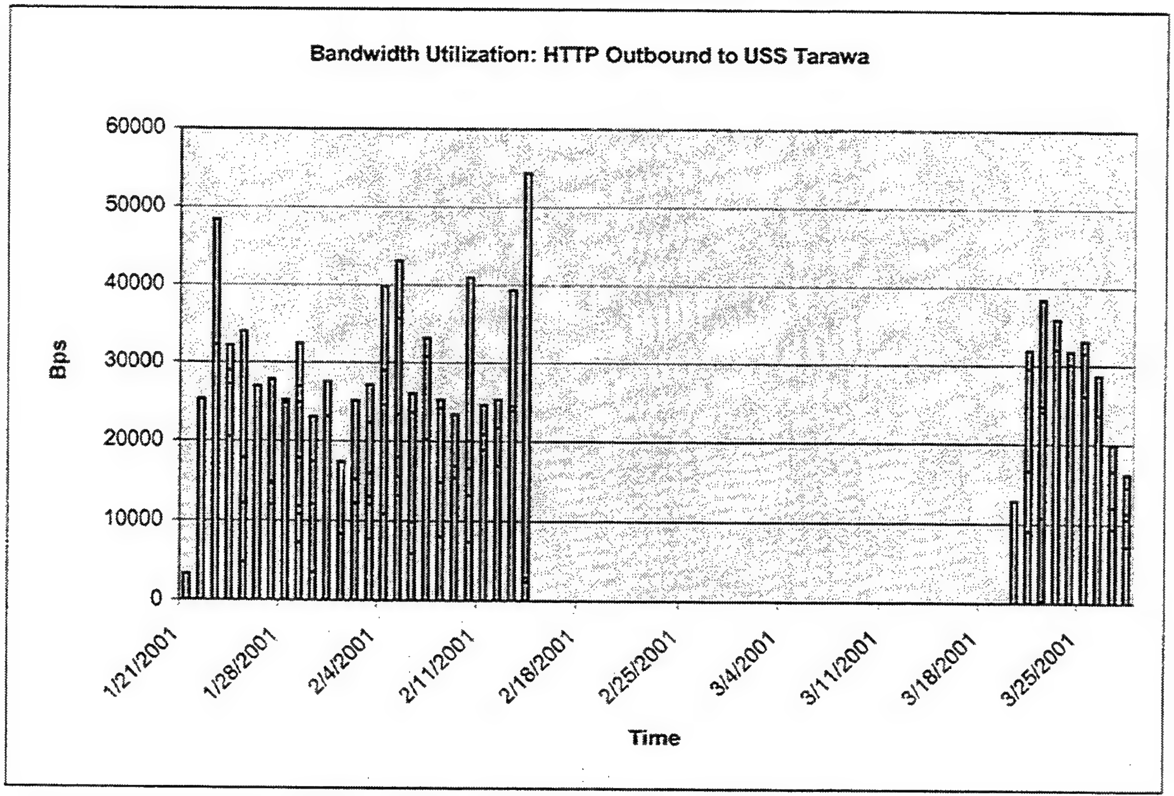


Figure A.3. HTTP Outbound to Tarawa.

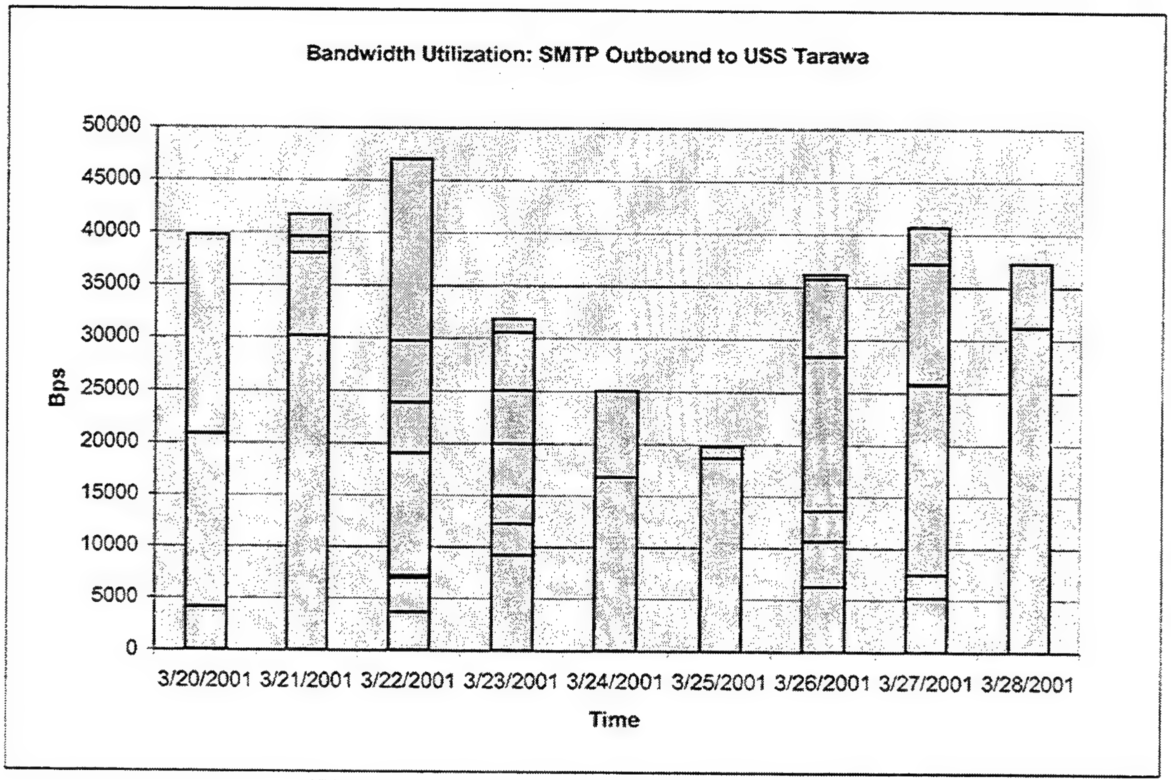


Figure A.4. SMTP Outbound to Tarawa.

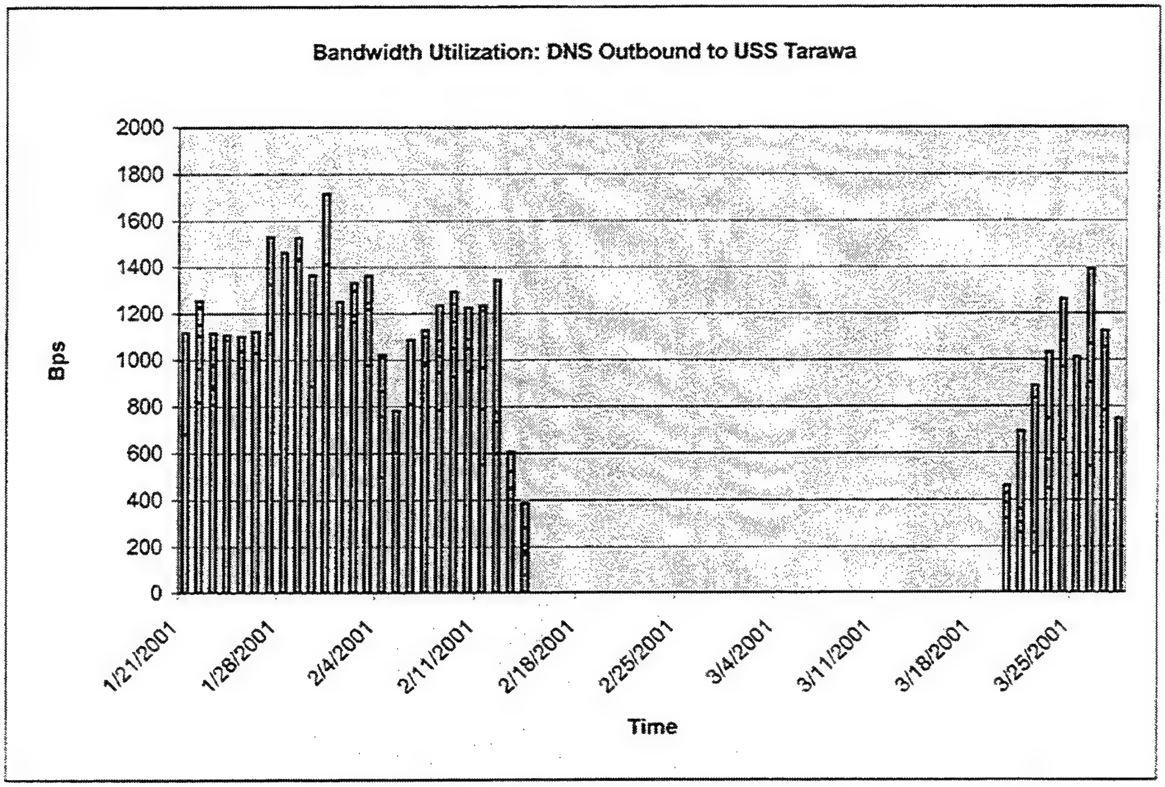


Figure A.5. DNS Outbound to Tarawa.

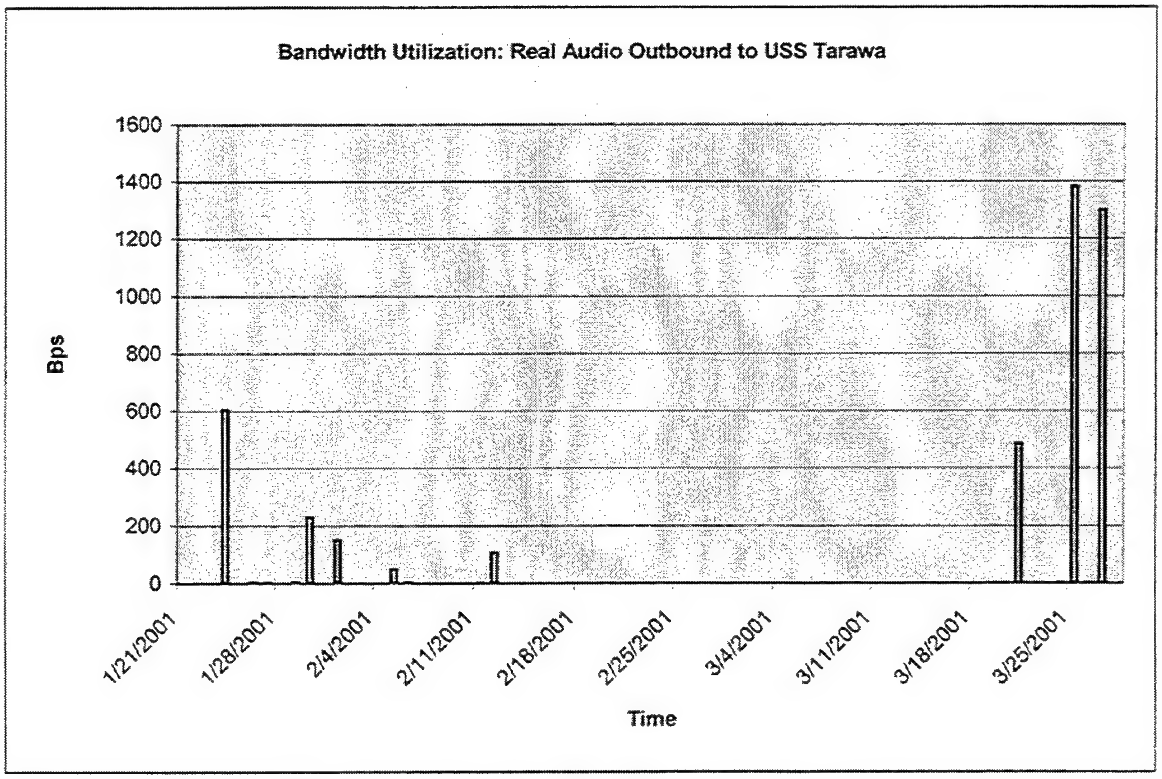


Figure A.6. Real Audio Outbound to Tarawa.

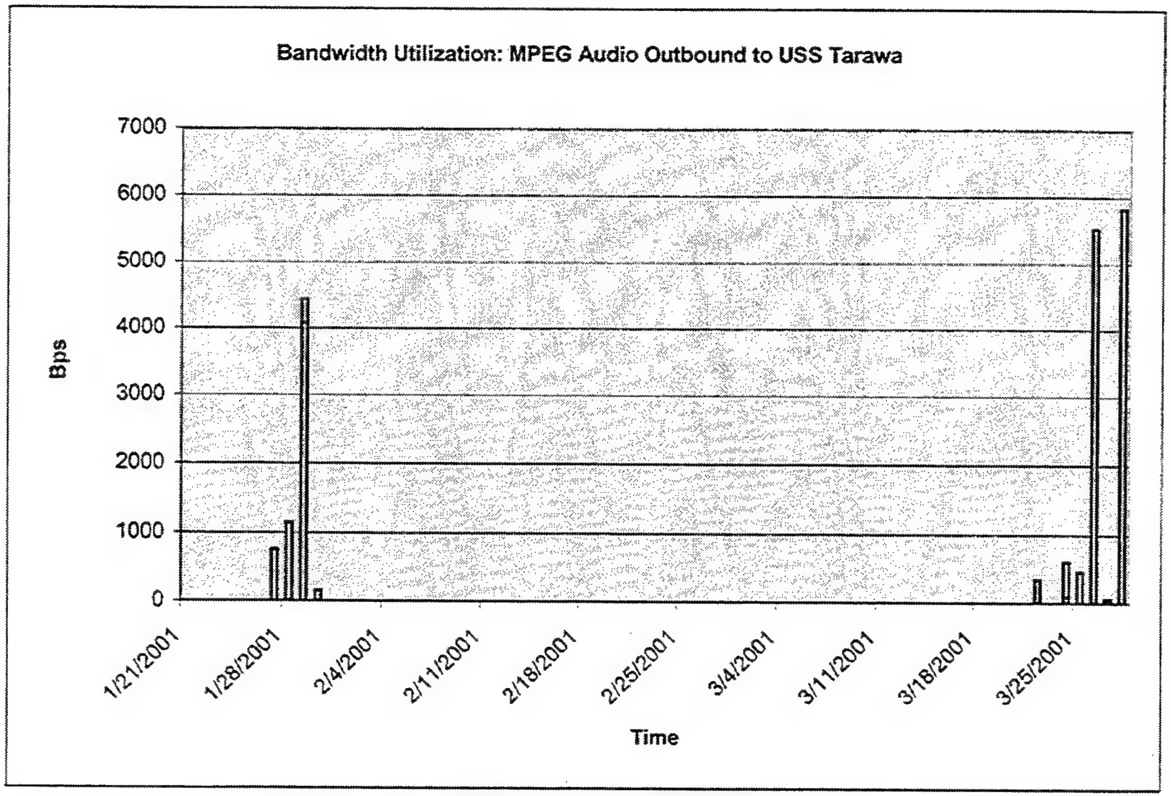


Figure A.7. MPEG Audio Outbound to Tarawa.

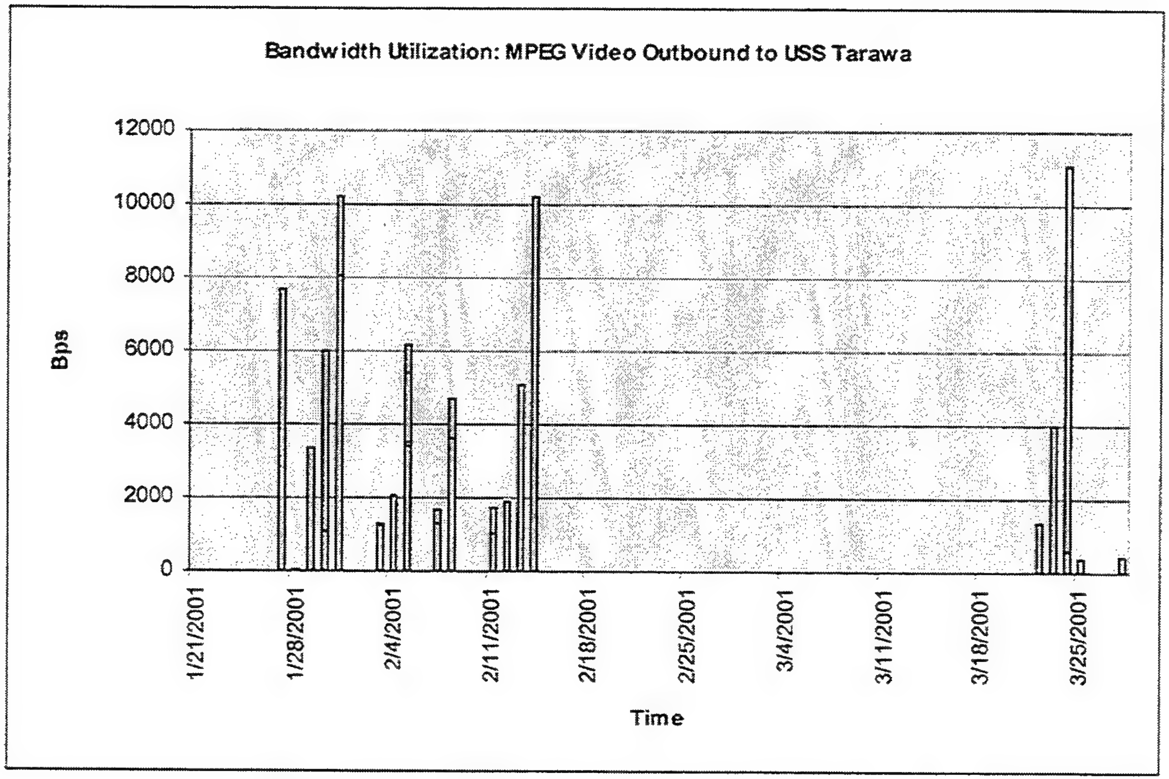


Figure A.8. MPEG Video Outbound to Tarawa.

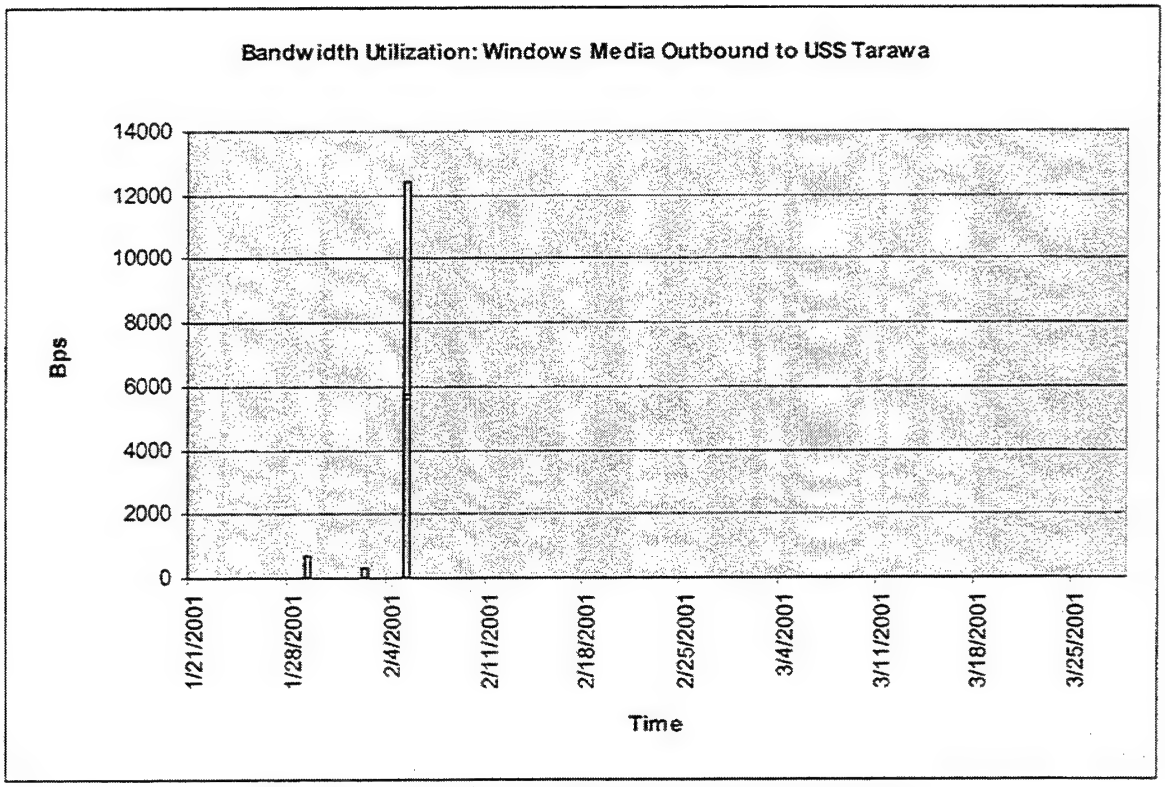


Figure A.9. Windows Media Outbound to Tarawa.

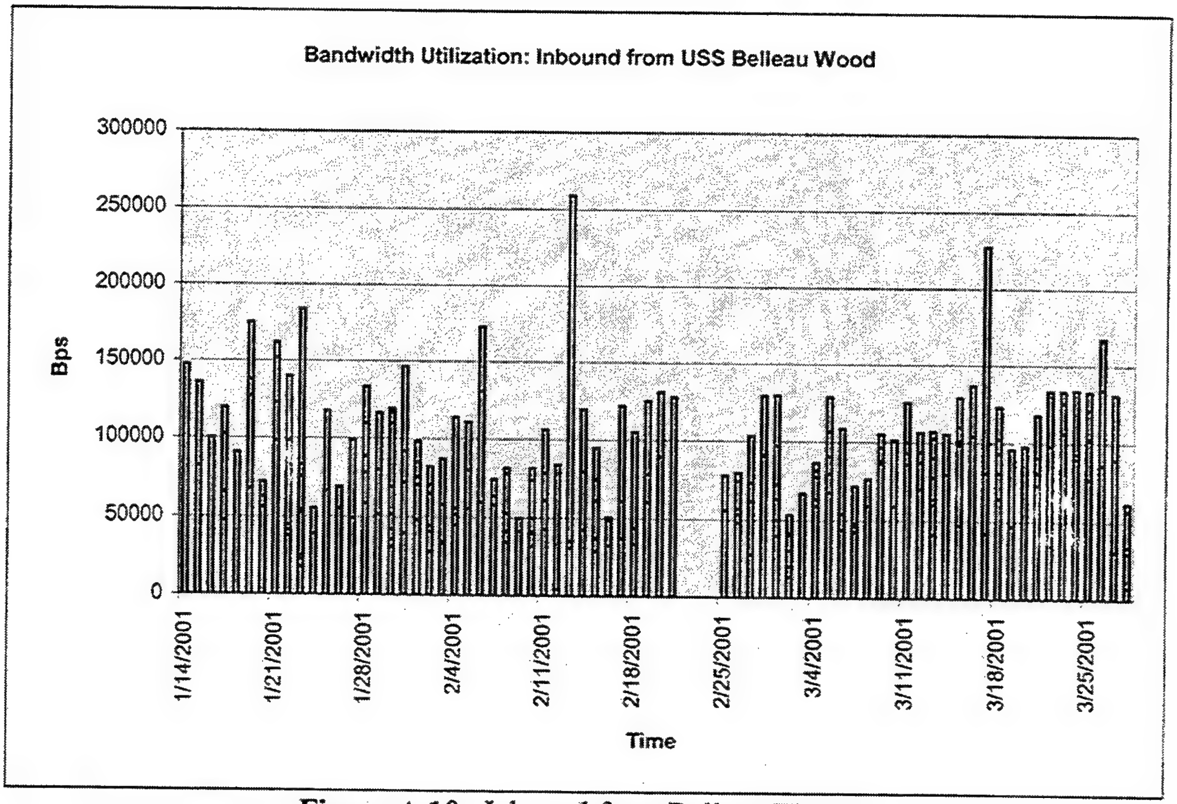


Figure A.10. Inbound from Belleau Wood.

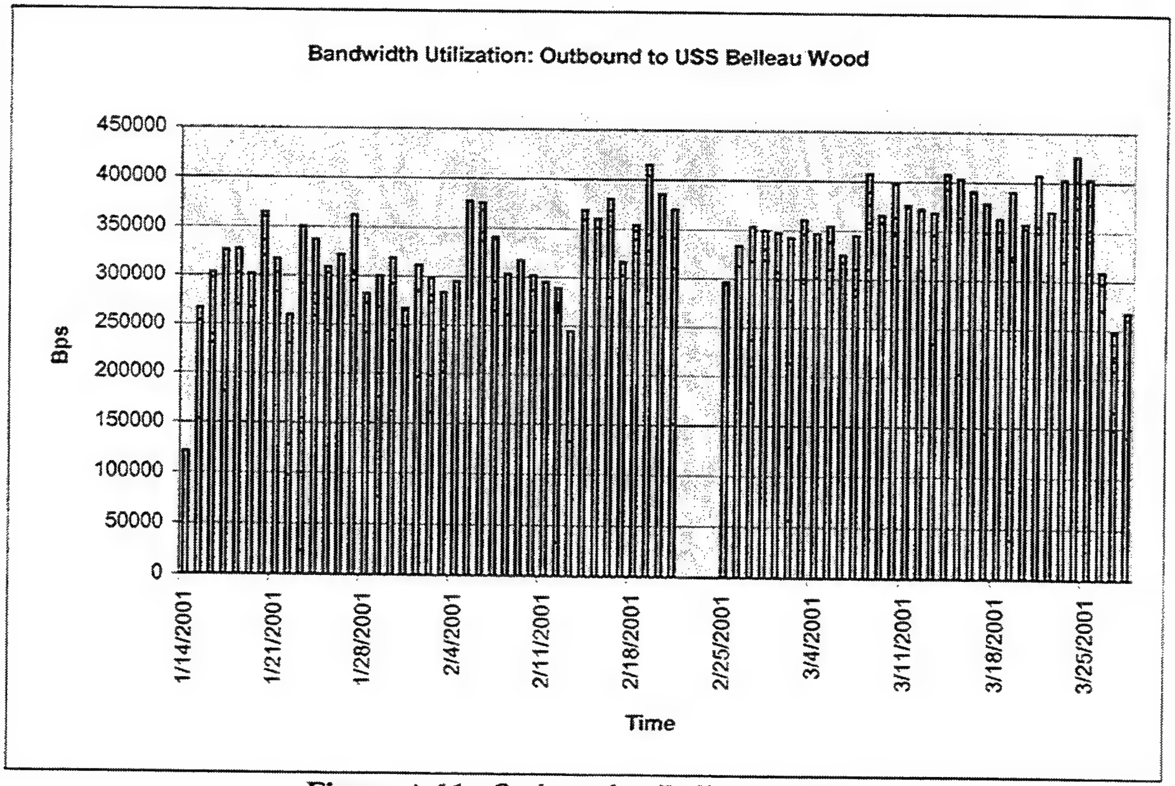


Figure A.11. Outbound to Belleau Wood.

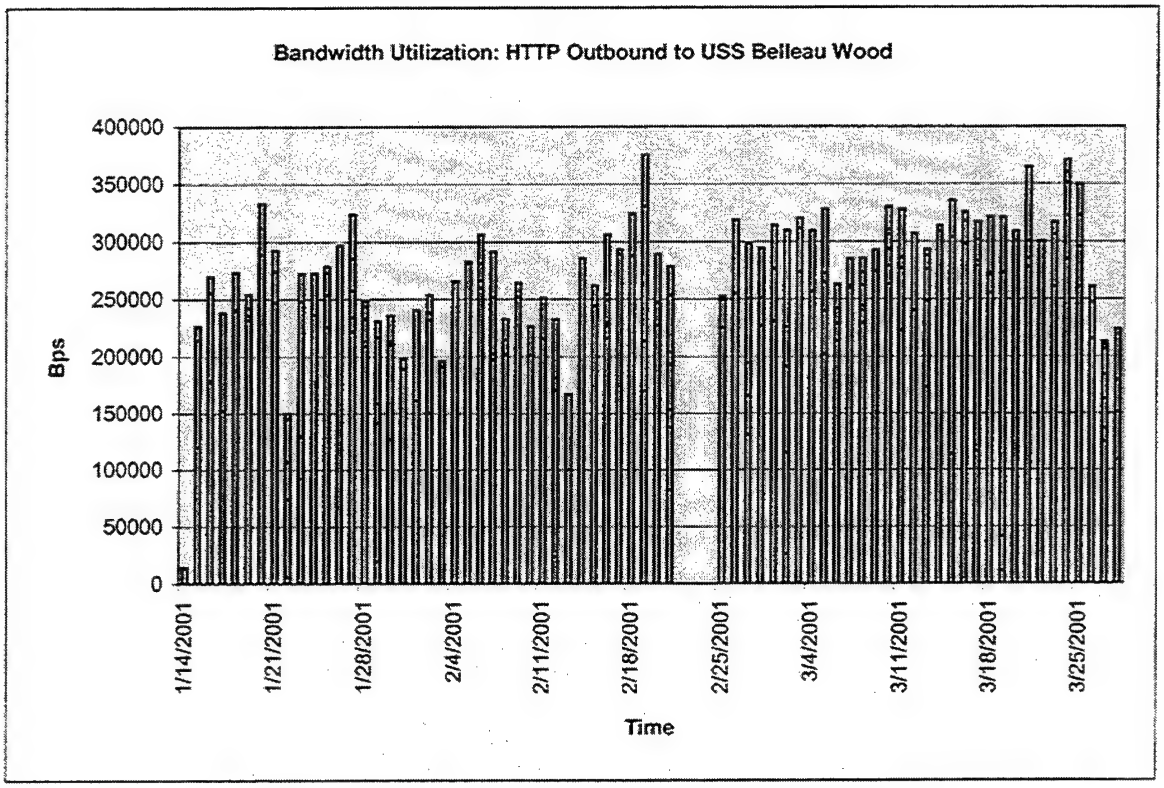


Figure A.12. HTTP Outbound to Belleau Wood.

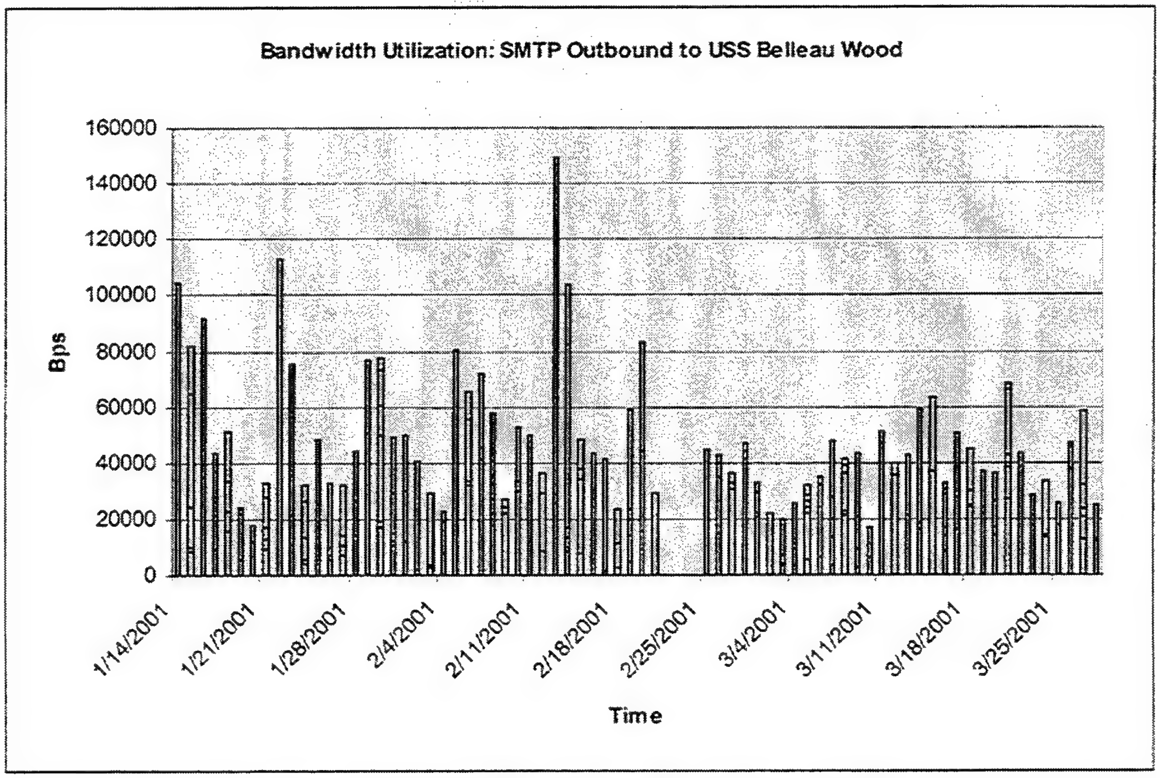


Figure A.13. SMTP Outbound to Belleau Wood.

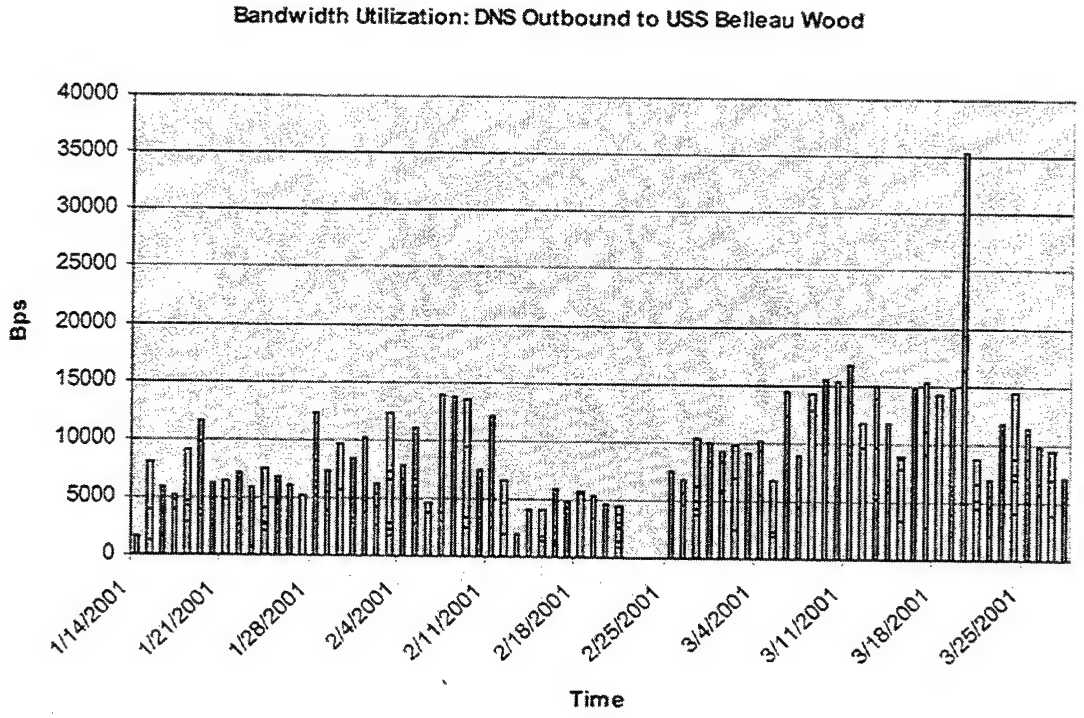


Figure A.14. DNS Outbound to Belleau Wood.

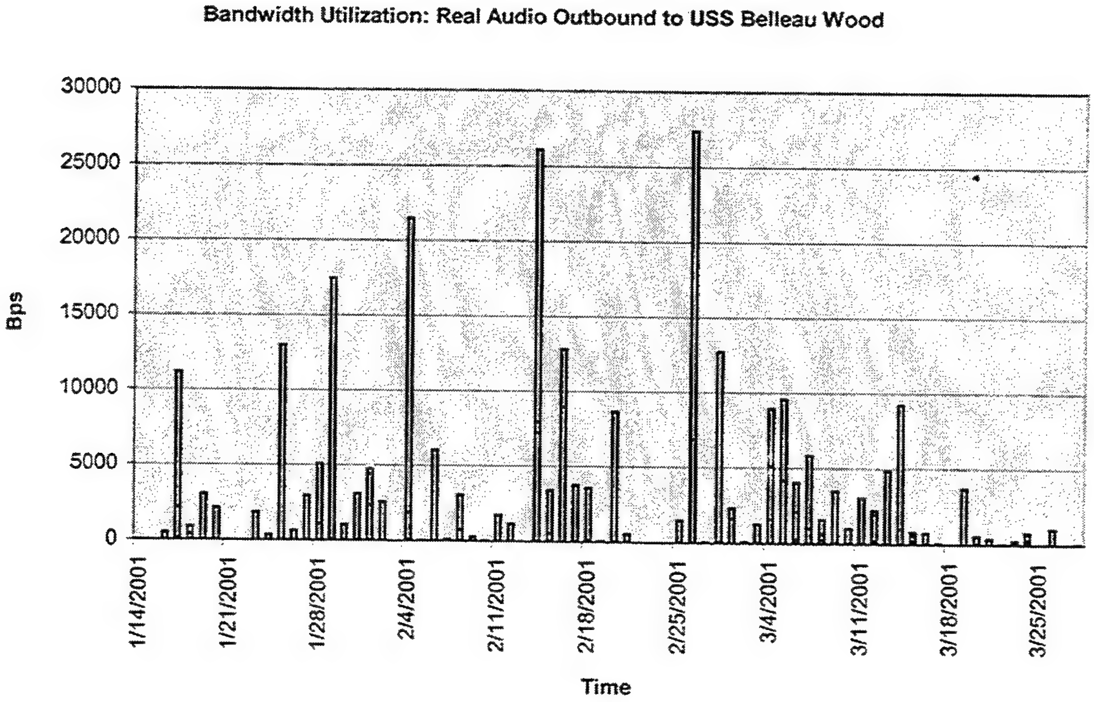


Figure A.15. Real Audio Outbound to Belleau Wood.

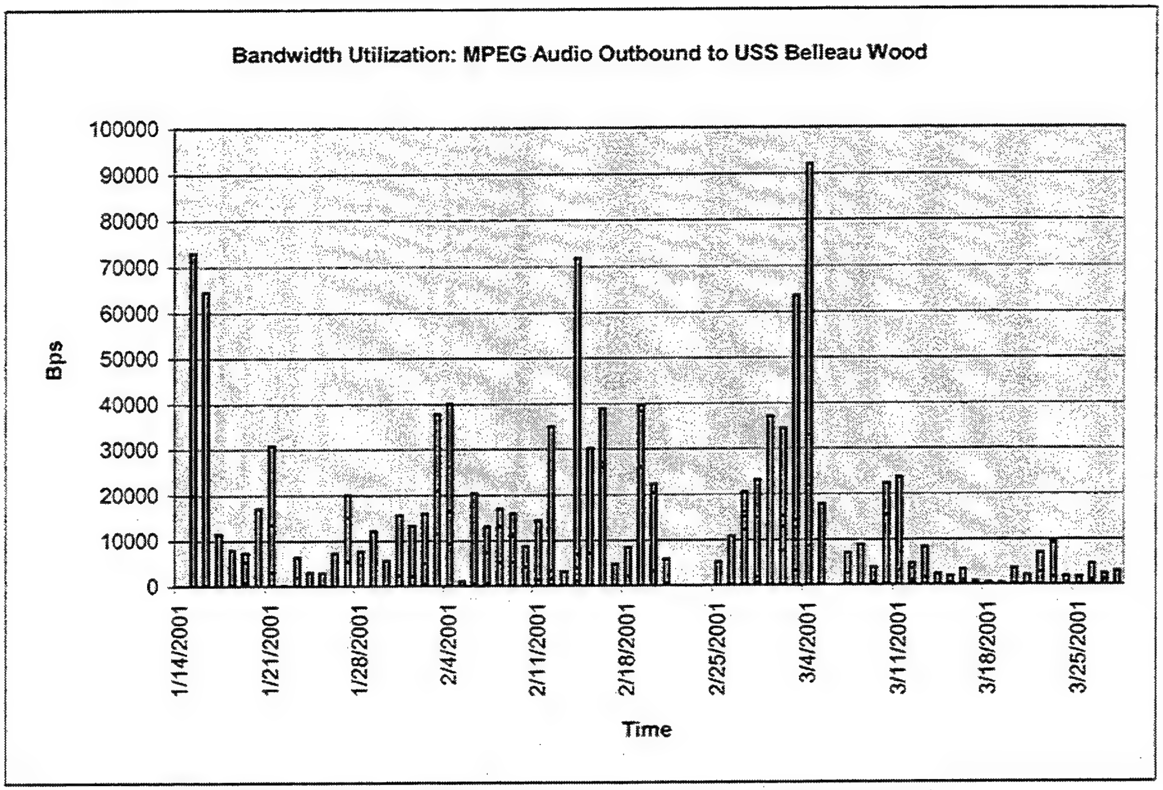


Figure A.16. MPEG Audio Outbound to Belleau Wood.

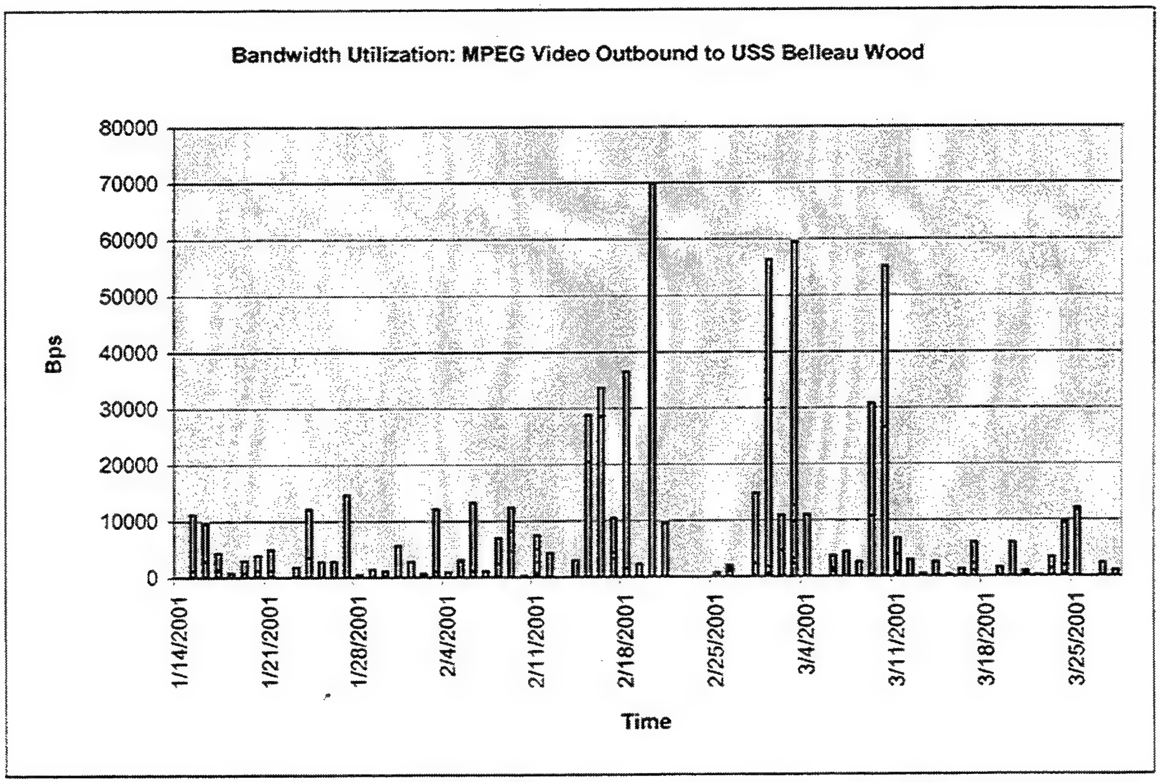


Figure A.17. MPEG Video Outbound to Belleau Wood.

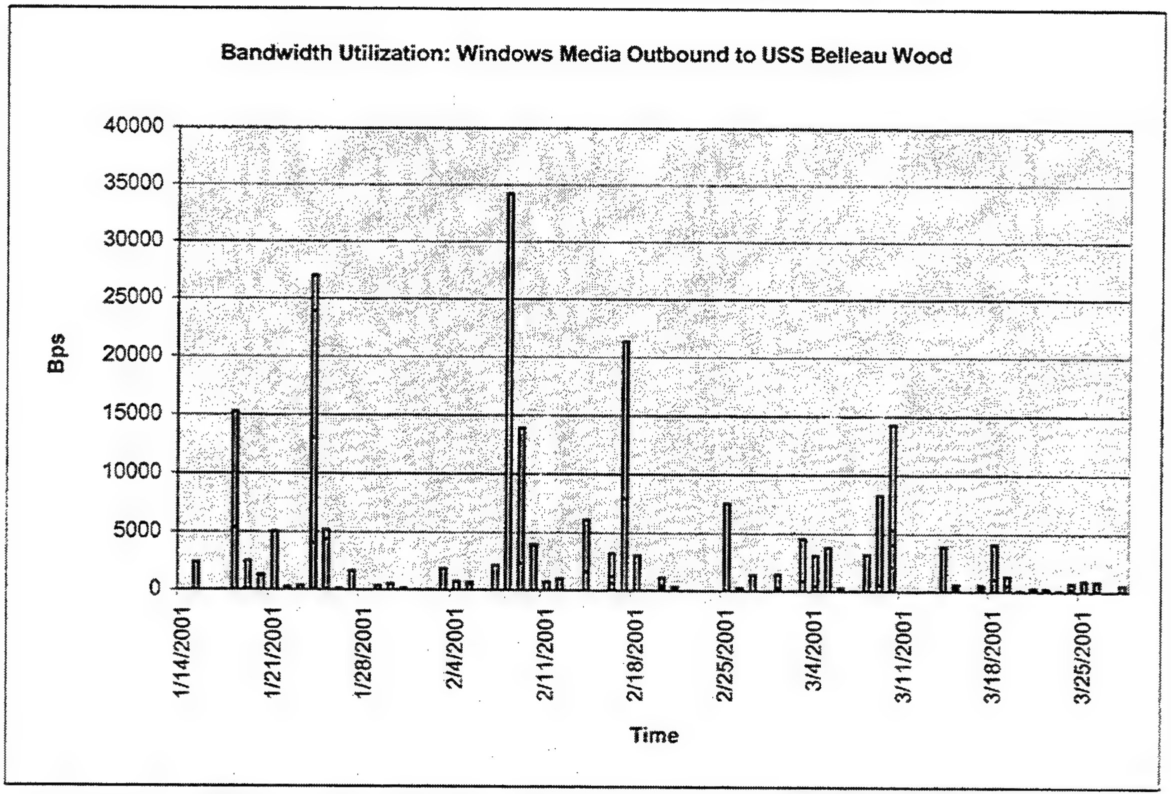


Figure A.18. Windows Media Outbound to Belleau Wood.

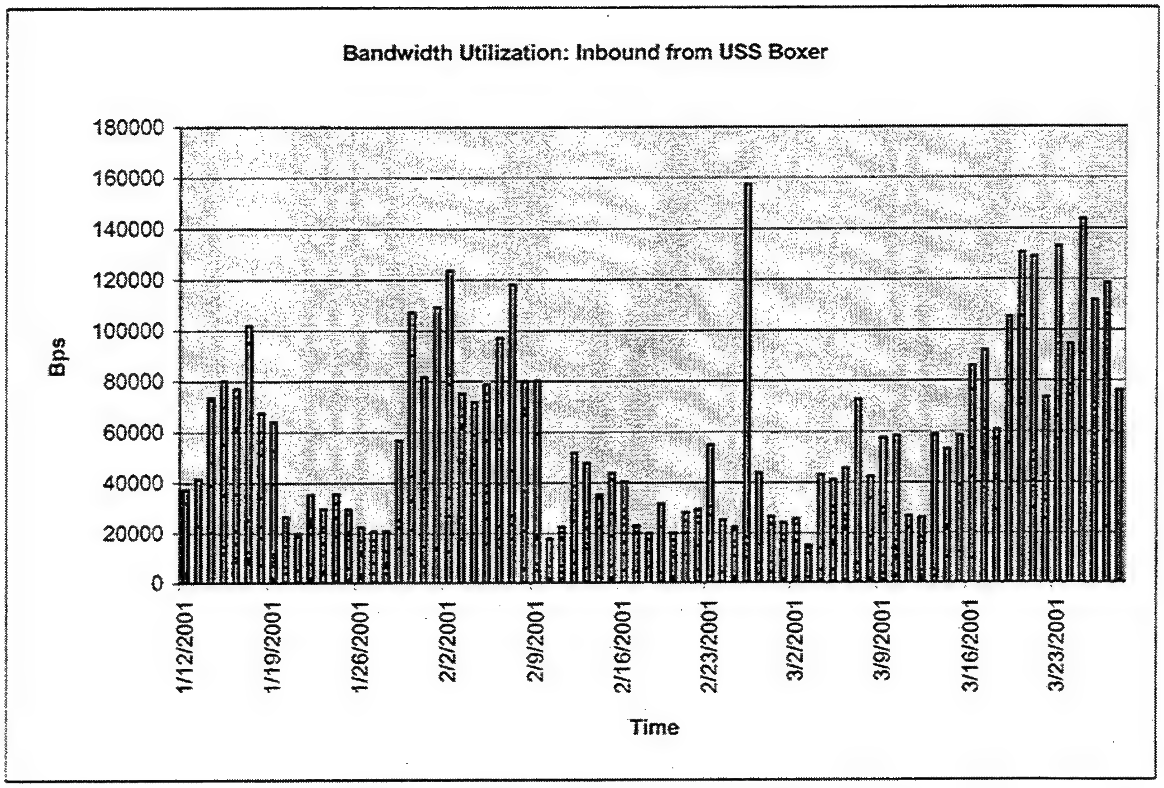


Figure A.19. Inbound from Boxer.

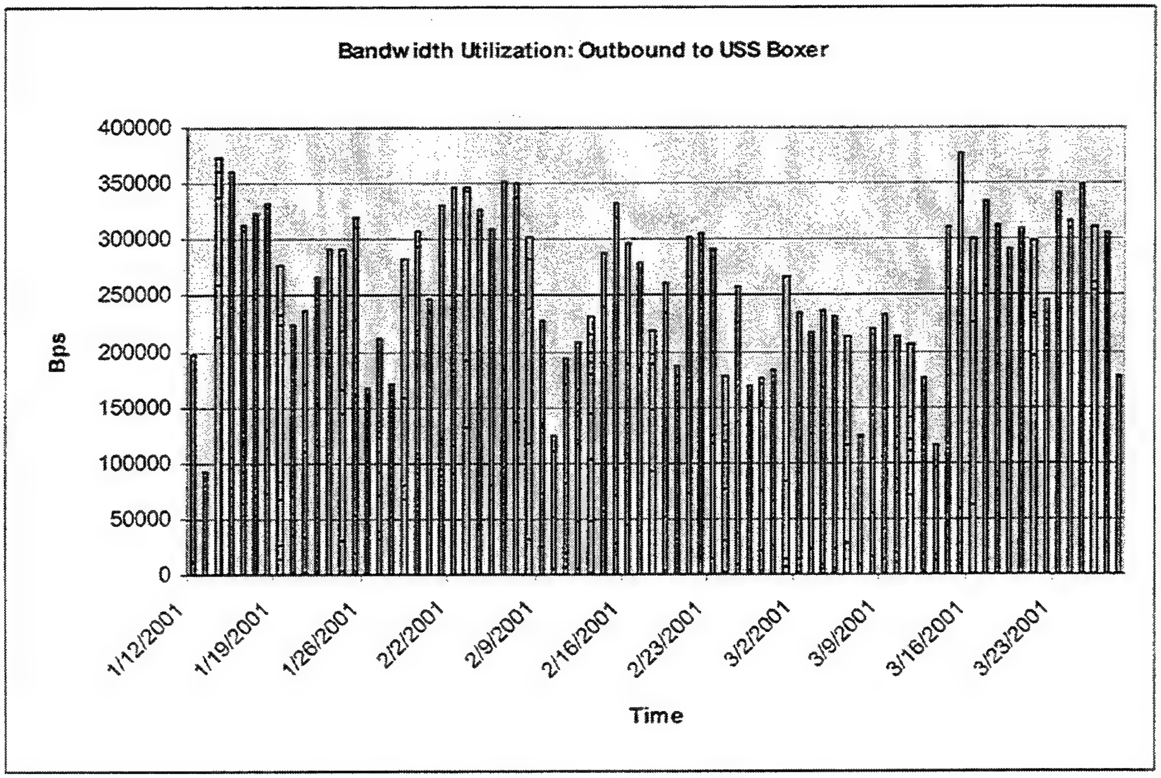


Figure A.20. Outbound to Boxer

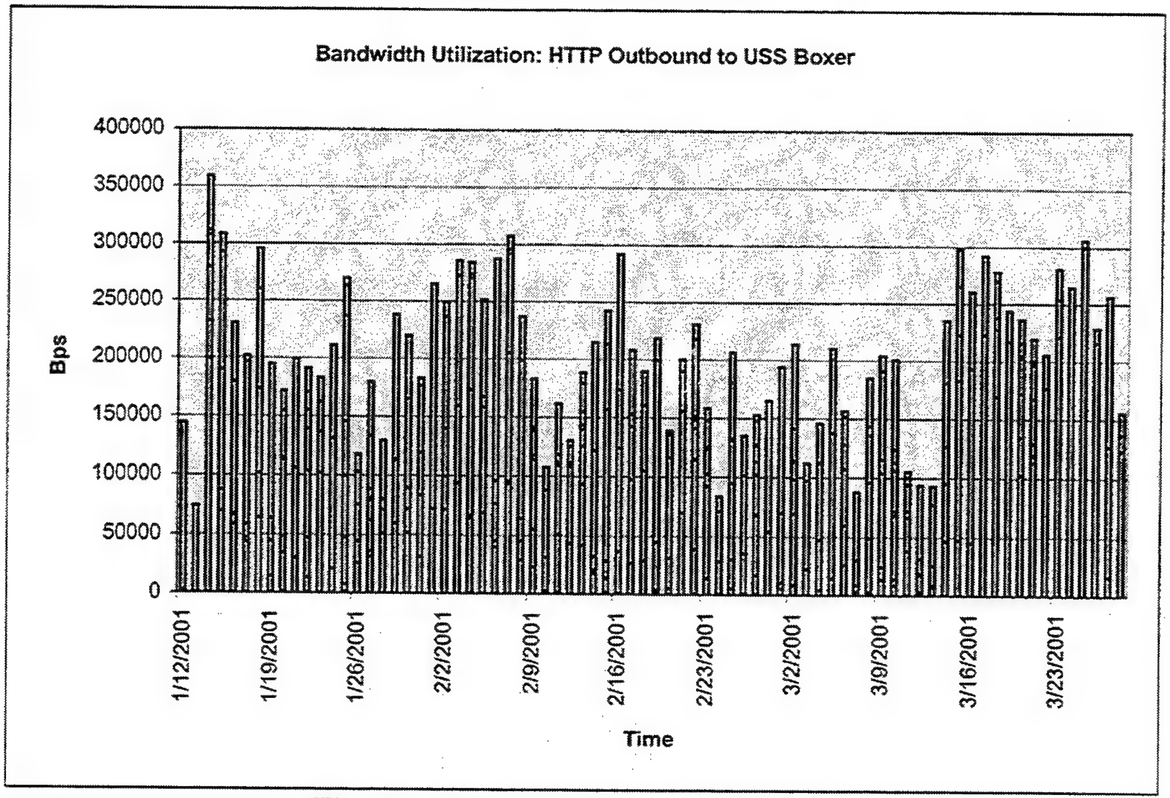


Figure A.21. HTTP Outbound to Boxer.

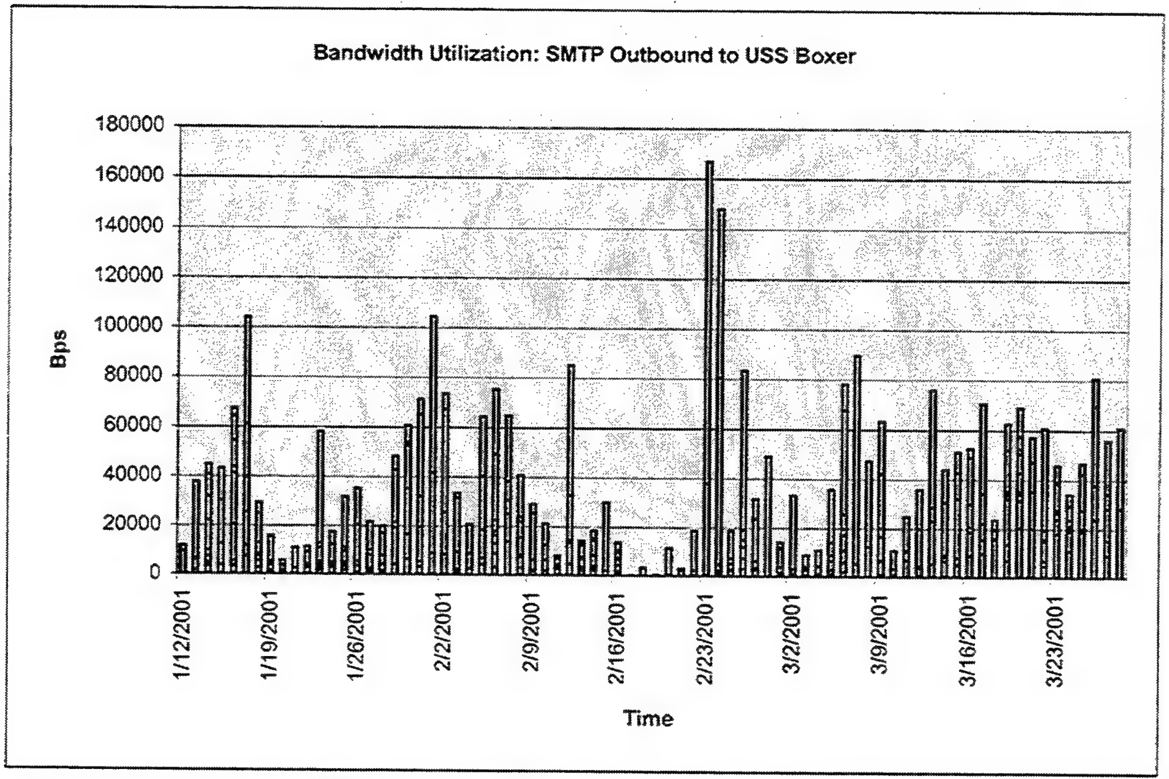


Figure A.22. SMTP Outbound to Boxer.

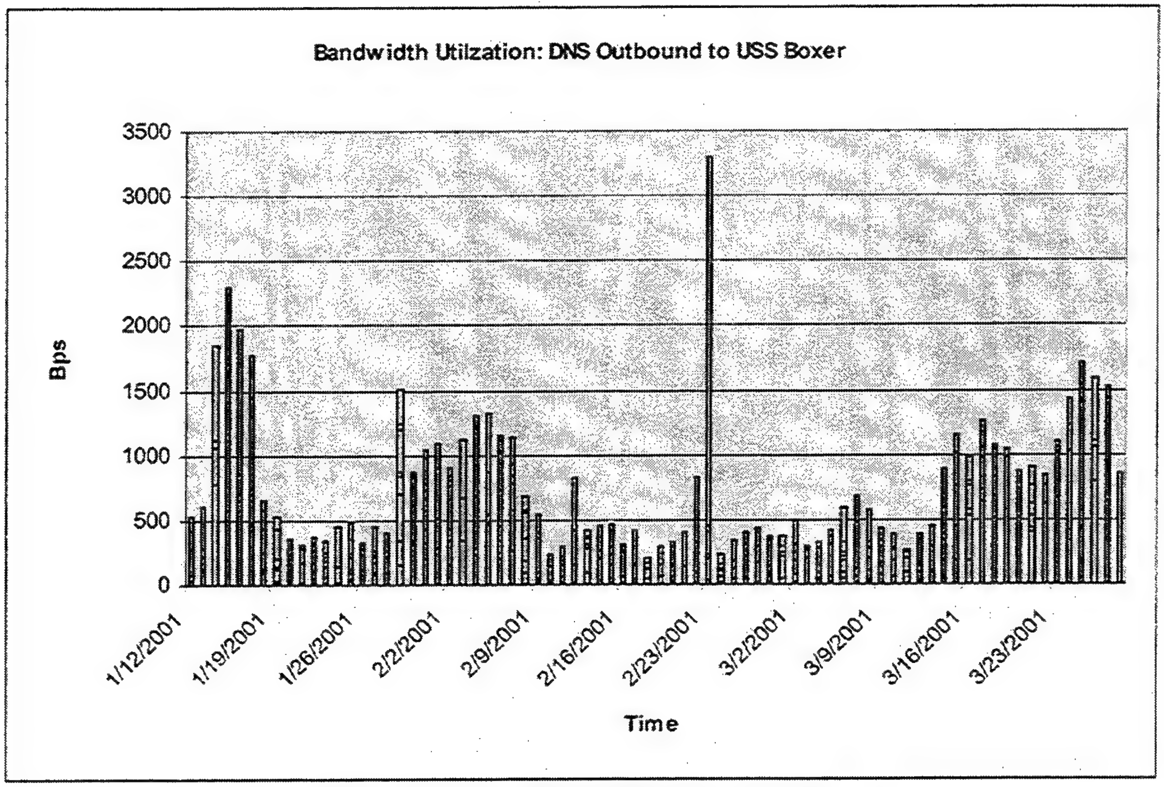


Figure A.23. DNS Outbound to Boxer

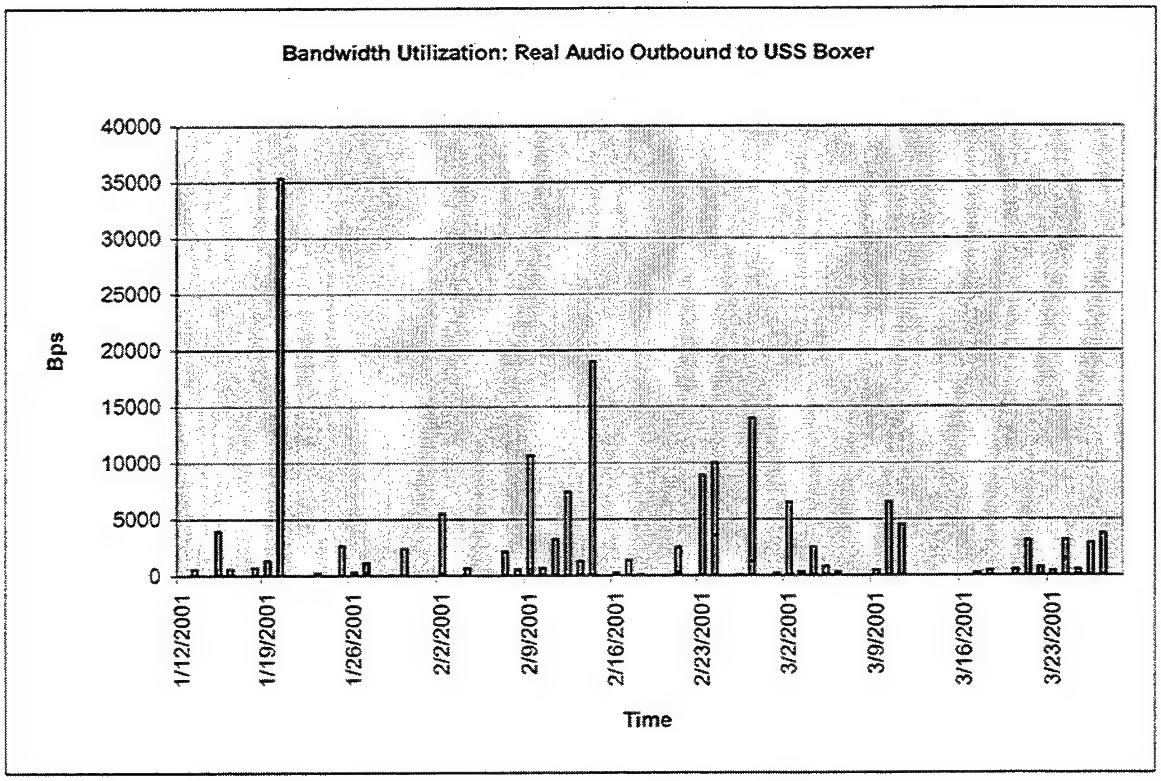


Figure A.24. Real Audio Outbound to Boxer.

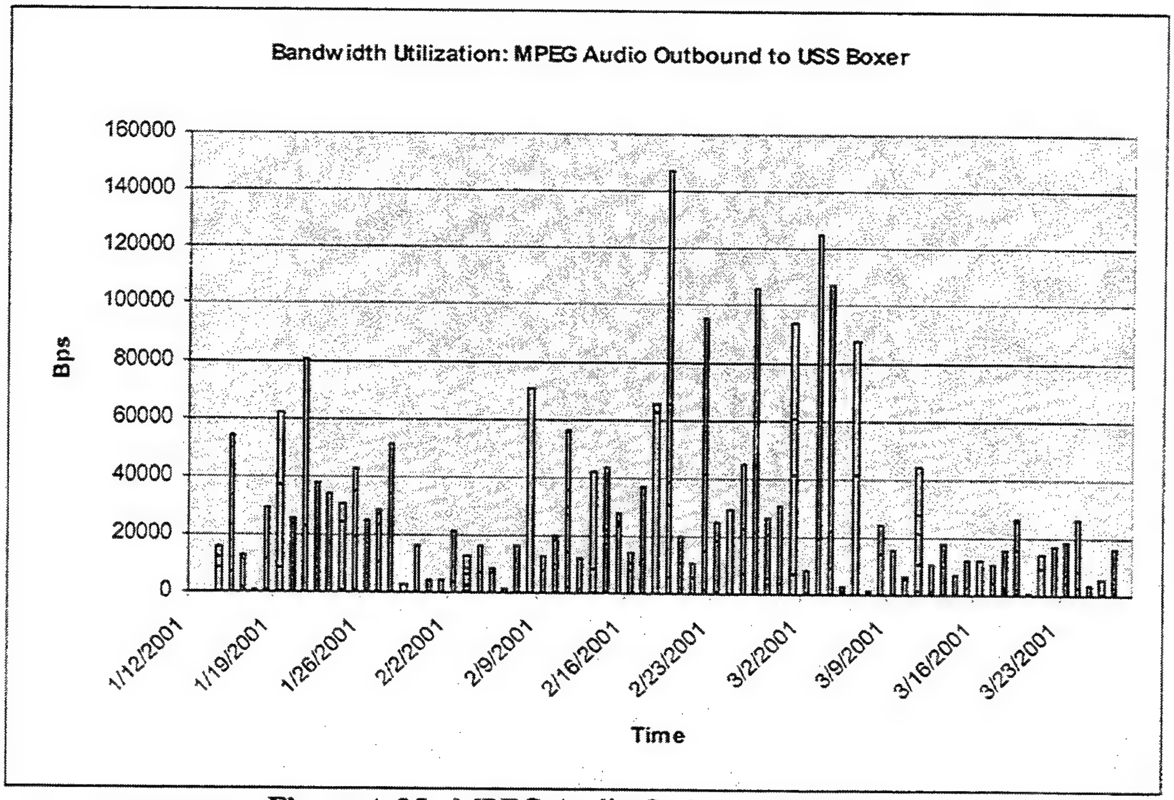


Figure A.25. MPEG Audio Outbound to Boxer.

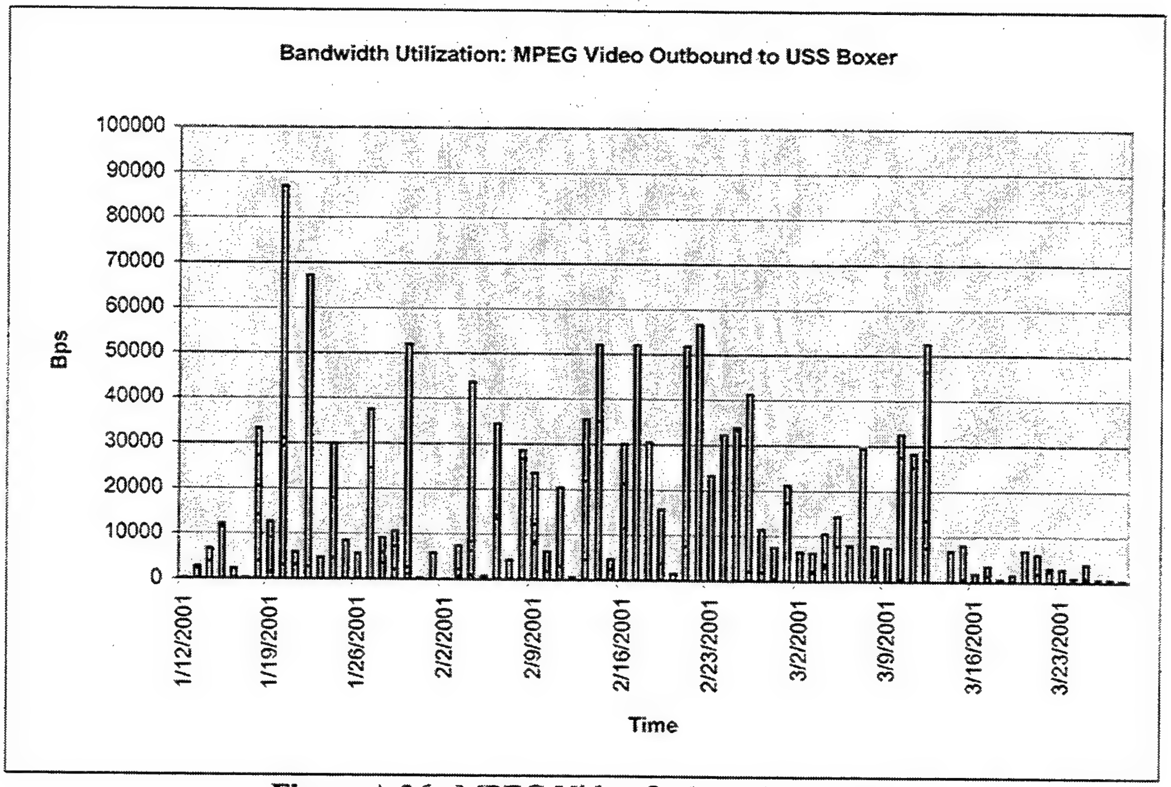


Figure A.26. MPEG Video Outbound to Boxer

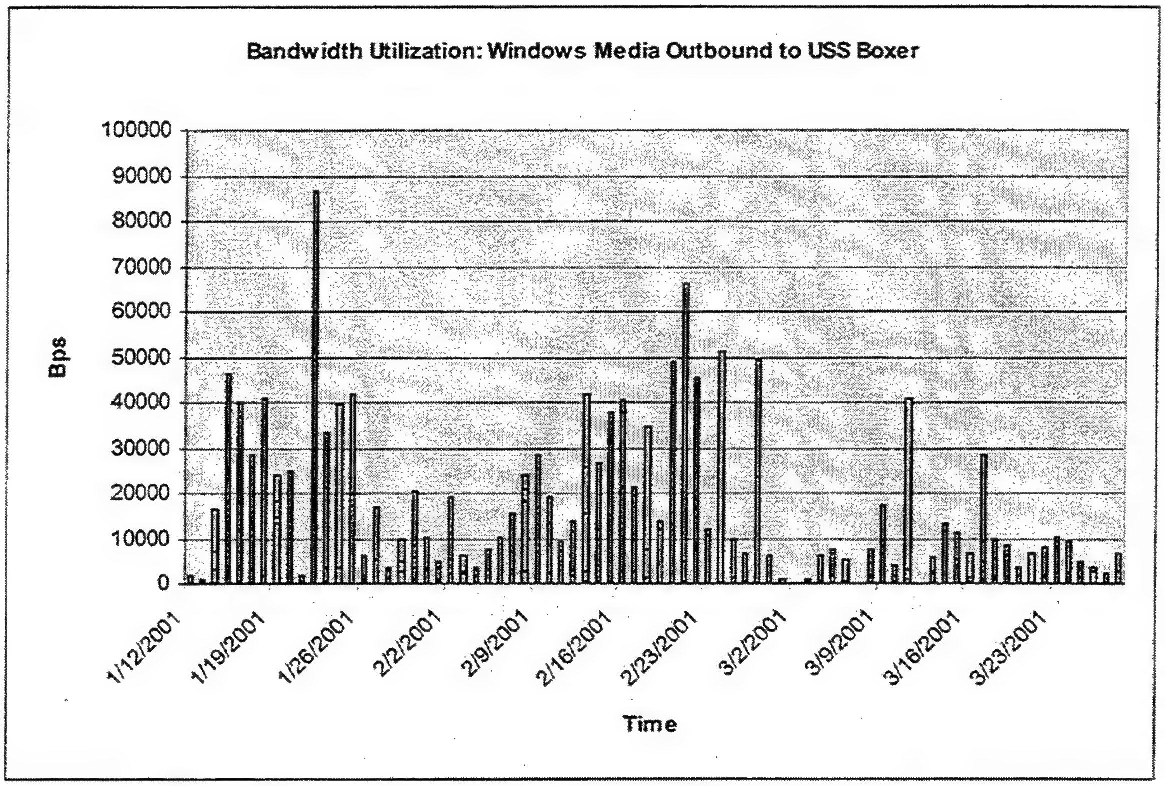


Figure A.27. Windows Media Outbound to Boxer.

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

"AgentBuilder: An Integrated Toolkit for Constructing Intelligent Software Agents," Revision 1.3, Reticular Systems, Inc., February 1999.

"Agent Technology Green Paper," Version 0.92, OMG Agent Working Group, April 2000.

Barneyback, S., "Command and Control: Relationship to Information Superiority and the C2 Core Tasks," Concept Paper, Joint Warfighting Center, Future Concepts Branch, 18 February 1998.

Bieszczad, A., et al, "Management of Heterogeneous Networks with Intelligent Networks," Bell Labs Technical Journal, pp. 109-135, October-December 1999.

Bigham, J. et al, "Multi-Agent System for Network Resource Management," Department of Electronic Engineering, Queen Mary and Westfield College, University of London, 1999.

Bissel, T. et al, "Service Level Management with Agent Technology," GMD-German National Research Center for Information Technology, TERENA Networking Conference, May 2000.

Bordetsky, A., "Adaptive QoS Management via Multiple Collaborative Agents," *Agent Technology for Communication Infrastructures*, Chapter 4, Wiley Books, 2001.

Bordetsky, A., Draft Proposal, Center for Research in Global Information Grid Operations (GRID Institute), Naval Postgraduate School, 2001.

Chairman, Joint Chiefs of Staff, "Enabling the Joint Vision," The Joint Staff, C4 Systems Directorate, Information Superiority Division (J6Q), May 2000.

Collis, J. and Ndumu, D., *The Zeus Agent Building Toolkit*: *Zeus Technical Manual*, Release 1.0, Intelligent Systems Research Group, BT Labs, September 1999.

Cuthbert, L., "Intelligent Agents for Managing Resources in Mobile Networks," Department of Electronic Engineering, Queen Mary and Westfield College, University of London, presented at the Naval Postgraduate School, 2 November 2000.

Dutta-Roy, A., "The Cost of Quality in Internet-style Networks," IEEE Spectrum, pp. 57-62, September 2000.

Gordon, D., "APT Agents: Agents That Are Adaptive, Predictable, and Timely," AI Center, Naval Research Laboratory, in the Proceedings of the First Goddard Workshop on Formal Approaches to Agent-Based Systems, 2000.

Hayzelden, A. and Bigham, J., "Agent Technology in Communications Systems: An Overview," Intelligent Systems Applications Group, Department of Electronic Engineering, Queen Mary and Westfield College, University of London, January 1999.

Hayzelden, A., Bigham, J., and Luo, Z., "A Multi-Agent Approach for Distributed Broadband Network Management," Intelligent Systems Applications Group, Department of Electronic Engineering, Queen Mary and Westfield College, University of London, 1999.

Kosuga, M. et al, "A Model of Adaptive QoS Management Platform Based on Cooperation of Layered Multi-Agents," Global Telecommunications Conference, GLOBECOM '99, Vol. 1A, pp. 406-413, 1999.

Lewis, L., *Managing Computer Networks: A Case-Based Reasoning Approach*, Artech House, Inc., 1995.

Lewis, L., "Spectrum Service Level Management Definition, Offerings, and Strategy," Cabletron Systems, March 1998.

Lewis, L., "Service Level Management with Spectrum," Aprisma Management Technologies, 2000.

Manola, F. and Thompson, C., "Characterizing the Agent Grid," Object Services and Consulting, Inc., June 1999.

Marsh, H., "Battlespace Dominance in Expeditionary Warfare: The Challenge of Defeating a Knowledge-Superior Enemy," Draft, Office of Naval Research, May 2000.

Mayer, R. and Stover, C., "The Application of Information Superiority to a Joint Task Force Headquarters," Joint Experimentation Directorate, U.S. Joint Forces Command.

McCabe, J., *Practical Computer Network Analysis and Design*, Morgan Kaufmann Publishers, Inc., 1998.

Nwana, H., "Software Agents: An Overview," Knowledge Engineering Review, Vol. 11, pp. 1-40, September 1996.

Nwana, H. and Ndumu, D., "An Introduction to Agent Technology," Applied Research andTechnology, BT Labs, 2001.

Odubiyi, J., "Old Wines in New Bottles: Evaluating Multi-agent Systems from Survivability Perspectives," Concert Networks and Systems Engineering, BT Labs, 2000.

Odubiyi, J. et al, "Proteus – Adaptive Polling Strategies for Proactive Management of ATM Networks Using Collaborative Intelligent Agents," BT North America Inc., presented at the 3rd International Conference on Autonomous Agents, Seattle, WA, May 1999.

OPNAV N6 Conference, Commander Seventh Fleet, 15 November 2000.

Packeteer's Packetshaper Performance Specifications, http: //www.packeteer.com/ products/packetshaper/index.cfm

"Policy Based Quality of Service for Enterprise LANs," Extreme Networks, Inc., 2000.

"A Primer on Policy-based Network Management," Hewlett-Packard Company, September 1999.

Sidor, D., "TMN Standards: Satisfying Today's Needs While Preparing for Tomorrow," IEEE Communications Magazine, pp. 54-64, March 1998.

Somers, F., et al, "Scalable Low-Latency Network Management Using Intelligent Agents," Broadcom Communicate, Vol. 3, Issue 2, 1998.

Stephenson, B. and Lee, W., *Concept of Operations for Fleet NOC Applications of Packeteer Packetshapers,* Pacific Region Network Operations Center, Wahiawa, Hawaii, 2001.

Sycara, K. et al, "Distributed Intelligent Systems," IEEE Expert, pp. 36-46, December 1996.

Vicente, J. et al, "Managing Enhanced Network Services: A Pragmatic View of Policy-Based Management," Intel Technology Journal Q1, pp. 1-13, 2000.

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center 2
   8725 John J. Kingman Rd., STE 0944
   Ft. Belvoir, VA 22060-6218

2. Dudley Knox Library 2
   Naval Postgraduate School
   411 Dyer Rd.
   Monterey, CA 93943-5000

3. Professor Alex Bordetsky, Code IS 1
   Naval Postgraduate School
   Monterey, CA 93943-5000

4. Professor John S. Osmundson, Code CC 1
   Naval Postgraduate School
   Monterey, CA 93943-5000

5. Professor Dan C. Boger 1
   Chairman, C3/IS Academic Groups
   Naval Postgraduate School
   Monterey, CA 93943-5000

6. Dr. Russel Richards 1
   U. S. Joint Forces Command
   Joint Futures Lab
   1562 Mitscher Ave., Suite 200
   Norfolk, VA 23551-2488

7. Mr. Dennis Popiela 1
   U. S. Joint Forces Command
   Joint Futures Lab
   1562 Mitscher Ave., Suite 200
   Norfolk, VA 23551-2488

8. Mr. William Zaldo 1
   U. S. Joint Forces Command
   Joint Futures Lab
   1562 Mitscher Ave., Suite 200
   Norfolk, VA 23551-2488

9. Mr. Ronald Adamo 1
U. S. Joint Forces Command
Joint Futures Lab
1562 Mitscher Ave., Suite 200
Norfolk, VA 23551-2488

10. Mr. Wrenwick Lee 1
NCTAMS PAC
500 Center St.
Wahiawa, Hawaii 96786

11. LT Raymond A. Rivera..1
4314 Corral Canyon Rd.
Bonita, CA 91902